

OHJELMISTOMETRIKKA TIETOJÄRJESTELMÄN TOIMITUKSEN JOHTAMISEN APUVÄLINEENÄ

Case Yritys X

LAHDEN AMMATTIKORKEAKOULU

Ylempi ammattikorkeakoulututkinto

Yrittäjäys ja liiketoimintaosaaminen

Opinnäytetyö

Kevät 2014

Arto Ihantoja

Lahden ammattikorkeakoulu

Yrittäjyys ja liiketoimintaosaaminen

IHANTOJA, ARTO: Ohjelmistometriikka tietojärjestelmätoimituksen johtamisen apuvälineenä

Yrittäjyyden ja liiketoimintaosaamisen ylemmän ammattikorkeakoulututkinnon
opinnäytetyö, 76 sivua, 41 sivua liitettä

Kevät 2014

TIIVISTELMÄ

Tutkimuksen tarkoituksena oli selvittää ohjelmistometriikan merkitys asiakaskohdaisesti räätälöidyn tietojärjestelmän kehittämisen sekä ylläpidon liiketoiminnan johtamiselle. Tutkimuskohteena oli tietojärjestelmätoimitus suomalaisessa IT-alan yritys X.

Kohdeyrityksessä oli tarve kehittää keinoja ennakoida sekä estää tietojärjestelmätoimituksien ohjelmistoteknisten laatupoikkeamien havaitsemista. Tutkimuksen tavoitteena oli kuvata kohdeyrityksen näkökulmasta ohjelmistometriikan merkitys, sijoittuminen kohdeyrityksen toimintamalliin ja selvittää ohjelmistometriikan nykytila.

Tutkimus pohjautui laadulliseen tutkimukseen hyödyntäen ohjelmistometriikan teorian sekä kohdeyrityksen toimintamalleja. Tutkimukseen sisältyi myös kohdeyrityksessä 2013 suoritettu kvantitatiivinen kyselytutkimus.

Tutkimuksen mukaan ohjelmistometriikan avulla on mahdollista vähentää laatu- ja kustannustekijöitä ohjelmistokehityksessä ja -ylläpidossa. Lisäksi ohjelmistometriikan avulla on mahdollista parantaa ohjelmiston objektiivista mittaamista sekä lisätä teknisen laadun läpinäkyvyyttä tietojärjestelmätoimituksen johtamiselle. Kyselytutkimuksen perusteella kohdeyrityksessä hyödynnettiin vaihtelevasti ohjelmistometriikkaa, mutta hyödyntämistapaa tulisi määrämuotoistaa. Tutkimuksen keskeinen lopputulos oli suositus kohdeyritykselle käynnistää ohjelmistometriikkaan liittyvä kehittämisprojekti.

Avainsanat: ohjelmistometriikka, mittaaminen, tietojärjestelmätoimitus, ohjelmistotuotanto, ohjelmistoliiketoiminta, johtaminen, apuväline

Lahti University of Applied Sciences
Master's Degree Programme in Entrepreneurship and Business Management

IHANTOJA, ARTO: Software metrics as tool to support management of
 information system delivery
 Case: Enterprise X

Master's Thesis in Entrepreneurship and Business Management, 76 pages, 41
pages of appendices

Spring 2014

ABSTRACT

This thesis studies software metrics to support information system development and maintenance management. The study was conducted for a Finnish IT-company X which had needs to develop new methods to predict and prevent quality deviations in information system delivery cases. The purpose of this study was to describe the meaning of software metrics for the case company, to find out how software metrics fit together with the process models already in place and to discuss the status of software metrics.

This study was based on qualitative research using theoretical sources and process models from the company. As a part of this study a quantitative online questionnaire survey was carried out during 2013.

Based on the results of this study, it seemed that using software metrics might lower the costs of quality in the field of software development and maintenance. Using software metrics improves the visibility of objective software measurement and the status of technical quality to support decision making. Based on the online questionnaire survey the usage of software metrics varied in the company and should be improved to a more systematic level. For the company this study recommends to start a development project to improve software metrics practices.

Keywords: software metrics, measurement, information system delivery, software business, management, tool

SISÄLLYS

1	JOHDANTO	1
1.1	Tutkimuksen tausta, tavoitteet ja rajaukset	2
1.2	Tutkimuksen teoreettinen viitekehys	4
1.3	Tutkimusmenetelmä	6
1.4	Tutkimuksen rakenne	7
2	OHJELMISTOMETRIKKA TIETOJÄRJESTELMÄN TOIMITUKSESSA	9
2.1	Ohjelmistometriikka ohjelmistokehityksen käytäntönä	9
2.2	Ohjelmistometriikka tietojärjestelmän toimituksen johtamisessa	12
2.3	Laatukustannustekijöiden minimointi ohjelmistometriikalla	14
2.4	Ohjelmiston lähdekoodin teknisen laadun merkitys	17
2.5	Ohjelmistokehityksen menetelmät, ympäristöt ja työvälineet	20
2.6	Ohjelmistometriikan hyödyntäminen	26
2.7	Ohjelmistometriikan hyödyntäminen toimitusta laajemmin	30
2.8	Ohjelmistometriikan jalkautukseen liittyviä muutostekijöitä	34
3	TIETOJÄRJESTELMÄTOIMITUS KOHDEYRITYKSESSÄ	37
3.1	Kohdeyrityksen mittaamiskulttuuri	37
3.2	Kohdeyrityksen tietojärjestelmän toimitusmalli	38
3.3	Kohdeyrityksen ohjelmistoprojektin toimitusmalli	41
3.4	Kohdeyrityksen tietojärjestelmätoimituksen Business Case	43
3.5	Kohdeyrityksen palveluiden tuottamisympäristöt	46
3.6	Ohjelmistometriikan hyödyntäminen kohdeyrityksen tietojärjestelmätoimituksessa	48
4	TUTKIMUSKONTEKSTI JA -MENETELMÄT	53
4.1	Tutkimuskonteksti	53
4.2	Tutkimusaineiston hankinta	53
4.3	Tutkimusaineiston käsittely ja analysointi	55
5	TUTKIMUKSEN TULOKSET	56
5.1	Kyselytutkimuksen tulokset	56
5.2	Kyselytutkimuksen tulosten analysointi	63
5.3	Tutkimuksen lopputulos ja toimenpidesuositus kohdeyritykselle	70

6	JOHTOPÄÄTÖKSET JA YHTEENVETO	75
	JULKAISTUT LÄHTEET	77
	JULKAISEMATTOMAT LÄHTEET	78
	LIITTEET	80

KUVAT

Kuva 1: Tutkimuksen rakenne.....	7
Kuva 2: Ohjelmistometriikka kokonaisuuden osana	10
Kuva 3: Ohjelmistoversioiden julkaisu MSF-prosessimallissa	22
Kuva 4: MSF vaiheistus ja päätason tehtävät	23
Kuva 5: Ohjelmiston käyttöympäristöt.....	24
Kuva 6: Visual Studio2013 Code Metrics Output View	25
Kuva 7: NDepend 5 Dashboard.....	26
Kuva 8: Projektin liittyvien arvioiden tekeminen.....	30
Kuva 9: Ohjelmistometriikan keruu, analysointi sekä hyödyntäminen.....	32
Kuva 10: Kohdeyrityksen johtamisen kolmio	37
Kuva 11: Kohdeyrityksen tietojärjestelmän toimitusmalli	39
Kuva 12: Kohdeyrityksen yksinkertaistettu projektimalli.....	41
Kuva 13: Kohdeyrityksen malli sopimuksen ja projektitoimituksen yhteydestä	42
Kuva 14: Kohdeyrityksen Business Case -malli	44
Kuva 15: Malli asiakaskohtaisen tietojärjestelmän toimituksesta.....	45
Kuva 168: Ohjelmistometriikka osana ohjelmistokehittäjän työtä.....	50
Kuva 17: Ohjelmistometriikan sijoittuminen kohdeyrityksen toiminnassa.....	70

TAULUKOT

Taulukko 1: Ohjelmistometriikan koulutustarve	56
Taulukko 2: Ohjelmistometriikan osaaminen ja koulutustausta.....	57
Taulukko 3: Osallistumishalukkuus ohjelmistometriikan koulutuksiin	58
Taulukko 4: Ohjelmistometriikan työvälinetilanne sekä hankintatarve	58
Taulukko 5: Ohjelmistometriikan ohjeisto	59
Taulukko 6: Ohjelmistometriikasta saatava hyöty	59
Taulukko 7: Ohjelmistometriikka tarjouspyynnöissä ja tarjousvalmistelussa.....	60
Taulukko 8: Ohjelmistometriikka tarjousvalmistelussa sekä estimoinnissa	60
Taulukko 9: Ohjelmistometriikan hyödyntäminen työssä ja xshoressa	61
Taulukko 10: Mittaamiskulttuurin nykytila.....	61
Taulukko 11: Ohjelmistometriikan kerääminen ja tiedon hyödyntäminen	62
Taulukko 12: Esimiehen antama tuki ja kannustus	62
Taulukko 13: Ohjelmistometriikan käyttöönoton esteet.....	63
Taulukko 14: Suositus ohjelmistometriikan kohderyhmästä.....	63

1 JOHDANTO

“Fools ignore complexity. Pragmatists suffer it. Some can avoid it. Geniuses remove it.”

Alan Perlis, American Scientist (Chambers & Associates Pty Ltd 2014.)

Ohjelmistometriikan soveltaminen johtamisen apuvälineenä on liiketoiminnallisesti sekä johtamisen näkökulmasta kiinnostava aihe, koska sen avulla on mahdollista parantaa ohjelmistoteknisen toteutuksen laatua sekä minimoida laatukustannustekijöitä liiketoiminnallisen tuloksen parantamiseksi. Ohjelmistometriikan soveltamisesta johtamisen apuvälineenä on saatavilla vähän tutkimuksia sekä soveltamismalleja, mitkä olisivat suunnattu liiketoiminnasta tai projektitoimituksista vastaaville. Tästä syystä aihepiirin tutkiminen arkipäivän hyödyntämisen kannalta tarjoaa uusia mielenkiintoisia tutkimuskohteita liiketoiminnan johtamisen näkökulmasta.

Lähtökohta tutkimukselle kohdeyrityksessä syntyi asiakaskohtaisesti räätälöidyn tietojärjestelmätoimituksen kriisiytymisestä. Tämän kriisin hoitamisen yhteydessä kohdeyrityksessä ilmeni tarve kehittää uusia keinoja ennakoida ja estää tulevaisuudessa vastaavat tietojärjestelmätoimituksen laatupoikkeamat, mikä on peruste tämän tutkimuksen tekemiselle.

Tein tutkimuksen **kohdeyritykselle x** (jatkossa kohdeyritys), joka on pitkäikäinen IT-alan suuryritys. Tutkimuksen aikana työskentelin kohdeyrityksessä esimiehenä, mikä mahdollisti minulle tutkimuskohteen tarkastelun kohdeyrityksen sisäisestä näkökulmasta.

1.1 Tutkimuksen tausta, tavoitteet ja rajaukset

Tutkimuksen perusajatuksena ja tulevaisuuden visiona oli luoda lähtökohta kohdeyrityksessä tehtävälle ohjelmistometriikan (software metrics) kehittämishankkeelle. Tutkimuksen päämääränä oli selvittää kohdeyrityksen lähtökohta ohjelmistometriikan hyödyntämisen kehittämiseksi osana arkipäivän ohjelmistoliiketoiminnan johtamista. Ensisijaisena tavoitteena oli luoda näkemys ohjelmistometriikan merkityksestä, hyödynnettävyydestä, vaikuttavuudesta sekä sijoittumisesta osaksi kohdeyrityksen toimintaa eli käytännössä perustella kehittämistarpeelle. Toissijaisena tavoitteena oli luoda käsitys kohdeyrityksen näkökulmasta ohjelmistometriikan hyödyntämisen edellytyksistä johtamisen apuvälineenä, jalkautuksen lähtökohdista sekä esteistä ohjelmistometriikkaan liittyvän kehittämishankkeen suunnittelun tueksi.

Keskeinen tutkimuskysymys:

Mitä hyötyä on ohjelmistometriikasta asiakaskohtaisesti räätälöitävän tietojärjestelmän kehittämisen ja ylläpidon liiketoiminnan johtamiselle kohdeyrityksessä?

Tutkimuskysymyksestä tarkentavat ala-tutkimuskysymykset olivat:

1. *Miten kohdeyrityksessä ohjelmistometriikkaa tulisi hyödyntää osana tietojärjestelmän myyntiä, kehittämistä ja ylläpitoa?*
2. *Mitkä kohdeyrityksen asiat mahdollistavat ohjelmistometriikan käyttöönoton ja soveltamisen osana arkipäivän työtä?*
3. *Mitkä kohdeyritykset asiat jarruttavat ohjelmistometriikan käyttöönottoa ja hyödyntämistä osana arkipäivän työtä?*
4. *Miten kohdeyrityksen ohjelmistotuotannon mittaamisen yhteyttä tulisi parantaa liiketoiminnan mittaamisen ja seurannan osalta?*

Ohjelmistometriikka on laaja sekä moniulotteinen aihealue. Tutkimuksen sisältö ja tarkastelunäkökulmat rajattiin kohdeyrityksessä seuraavasti:

1. Suomen maayhtiön yksi liiketoimintayksikkö
2. ohjelmistoliiketoiminta
3. asiakaskohtaisesti räätälöidyn tietojärjestelmän myynti, toimitus ja ylläpito
4. asiakaskohtaisesti räätälöidyn tietojärjestelmän toimitus projektina

5. toimittajan näkökulmasta tietojärjestelmän elinkaaren keskeiset vaiheet: hankintailmoitus, tarjouspyyntö, tarjous, sopimus, toimitus, käyttöönotto, ylläpito, alasajo
6. huomioda kevyesti kohdeyrityksen näkökulmasta onshore, nearshore, offshore, blended shore toimituksia edellä mainittuja rajoituksia noudattaen

Tutkimuksen sisältöön tai tuloksiin eivät sisällyneet johtaminen, johtamisjärjestelmä, muutoshallinta, projektitoiminta, kohdeyrityksen toimintamallin kehittäminen tai jalkauttaminen. Ohjelmistometriikan osalta osalta tutkimukseen ei sisällynyt ohjelmistometriikkaan liittyvien mittareiden teorial, toimintaperiaatteet tai niiden käytännön selvittäminen.

Tutkimus tuotti lopputuloksena jatkotoimenpide-ehdotuksen kohdeyrityksen liiketoimintayksikön johtoryhmälle ohjelmistometriikan hyödyntämiseksi osana operatiivista johtamista. Tutkimuksen lopputuloksesta saavutettava hyöty kohdeyritykselle muodostui päätöksenteon tuesta kehittämissuunnitelman suunnittelemiseksi sekä käynnistämiseksi. Tutkimuksen lopputuloksen kohderyhmä oli kohdeyrityksen ohjelmistoliiketoiminnan tulosvastuulliset johtajat sekä esimiehet.

Tutkimuksen työhypoteesina oli, että tutkimuksesta saatavat hyödyt mahdollistavat kohdeyritykselle keinot:

- parantaa tietojärjestelmään liittyvän teknisen toteutuksen laatua sekä minimoida ohjelmistotoimituksen laatu- ja kustannustekijöitä
- lisätä tietojärjestelmän teknisen laadun läpinäkyvyyttä ja tilannetietoa päätöksenteon tueksi
- ennakoita ikäviä yllätyksiä ajoissa tehostamalla proaktiivista toimintaa

Tutkimus oli osa laajempaa kohdeyrityksen kehittämisen työkokonaisuutta. Ohjelmistometriikan soveltamisen määrämuotoistaminen johtamisen apuvälineenä tähtäsi konkreettisten arkipäivän ohjelmistokehityksen tavoitteiden, mittaamisen, seurannan, korjaavien toimenpiteiden suorittamisen ja näihin liittyvien keinojen kehittämiseen.

1.2 Tutkimuksen teoreettinen viitekehys

Ohjelmistometriikasta oli saatavilla vaihtelevissa määrin materiaalia, tutkimuksia sekä raportteja. Saatavilla oleva aineisto oli pääasiassa englanniksi. Aihepiiriin liittyvää suomenkielistä aineistoa oli niukasti saavilla. Esimerkiksi Googlen hakukone löysi englanniksi paljon hakutuloksia, mutta suomeksi vähän. Toisena esimerkkinä mainittakoon, että Googlen hakukone löysi hakusanoilla ”ohjelmistometriikka” ja ”johtaminen” ainoastaan Harri Pirittisen Six Sigma ohjelmistokehityssä -gradun. Ohjelmistometriikan tutkimukset olivat ohjelmistotekniikkalähtöisiä ja suunnattu enimmäkseen ohjelmistokehittäjille tai tutkijoille. Lisäksi saatavilla oleva aineisto oli vaikeaa ymmärtää ja soveltaa ilman riittävää ohjelmistotekniikan osaamista.

Ohjelmistometriikka on laaja osa-alue, mihin liittyy vaihtelevia koulukuntia, käytäntöjä sekä näkemyksiä. Ohjelmistometriikan käytännöt ovat alalla vielä vakiintumatta. ISO-standarointijärjestö ei ole vielä tuottanut ohjelmistometriikkaan liittyviä ISO-standardeja. ISO 9126-1 ohjelmiston laatuun liittyvän standardin osioissa 2 - 4 on kuvattu ISO teknisten raporttien tasolla yhteenä noin 250 metriikkaa. (Abran 2010, 6.)

Maailmalta löytyy ohjelmistotuotantoon liittyvää mittaustietoa kerääviä sekä tilastoisia organisaatioita. Esimerkiksi ISBSG (ISBSG 2012) tuottaa kerättyä kokeuksellista mittaustietoa tukemaan projektien laajuuden mitoittamista esimerkiksi toimintopistelaskennan (Function Points) muodossa. Toisena esimerkkinä ITMPI (ITMPI 2012) pyrkii edistämään ohjelmistoalan tuottavuutta tuottamalla raportteja sekä seminaareja aiheeseen liittyen. Suomessa asiaa edistää Finnish Software Measurement Association (FiSMA ry 2014).

Tutkimuksen haasteellisuutta lisäsi vaatimus tarkastella ohjelmistometriikkaa kohdeyrityksen sekä sen käytäntöjen näkökulmasta. Edellä mainituista syistä johtuen tutkimuksen aihepiiriin mukaista aineistoa ohjelmistometriikan hyödyntämisessä ohjelmistoliiketoiminnan johtamisen apuvälineenä ei ollut saatavilla.

Tutkimuksessa hyödynnettävä teoreettinen viitekehys ja siihen liittyvä keskeinen tietoperusta muodostui seuraavista osa-alueista:

1. Ohjelmistometriikka
2. Mittaaminen

Ohjelmistometriikan sekä ohjelmistotuotannon osalta keskeisimmän teoreettisen tietoperustan muodostivat seuraavat lähteet:

1. *Software Metrics and Software Metrology*, kirjoittaja tohtori Alain Abran on toiminut tietojärjestelmien sekä ohjelmistokehittämisen parissa kaksikymmentä vuotta. Abran on Quebecin yliopiston Software Engineering Research Laboratoryn johtaja. Hän on osallistunut myös ohjelmistokehityksen ISO/IEC JTC 1/SC 7 -standardointiin keskeisessä roolissa.
2. *The Economics of Software Quality*, teoksen kirjoittaja Jones Capers on yksi ohjelmistokehityksen keskeisimmistä vaikuttajista ja toimii Software Productivity Research LLC:ssä päättäjänä. Teoksen toinen kirjoittaja, Oliver Bonsignour, on toiminut tietohallintojohtajana Ranskan puolustusministeriön Advanced Research -yksikössä.
3. *Software Maintenance Management - Evaluation and Continuous Improvement*, kirjoittaja professori Alain Abran sekä Quebecin yliopiston professori April Alain, joka on tuottanut sisältöä ohjelmiston sisäiseen mittaamiseen osaksi artikkelia ISO 9126-3.
4. *Linda Westfall* on toiminut merkittävässä roolissa 30 vuoden ajan ohjelmistokehityksen, laadun ja metriikan tiimoilta. Westfall on toiminut esimerkiksi American Society of Qualityn puheenjohtajana.
5. *Software Estimation – Demystifying the Black Art*, useasti palkittu kirjoittaja Steve McConnell on tuottanut monia ohjelmistokehitykseen liittyviä arvostettuja teoksia.
6. *Clean Code - a handbook of Agile Software Craftsmanship*, kirjoittaja Robert C. Martin on ohjelmistokehityksen tunnustettuja auktoriteetteja maailmanlaajuisesti useilla osa-alueilla.

Hyödynsin tutkimuksessa kohdeyrityksen sähköisen toimintajärjestelmän tarjoamia prosessikuvauksia sekä ohjeita tutkimuksen rajauksen mukaisesti. Sovelsin kohdeyrityksen toimintajärjestelmän sisällöstä aihe-alueita: mittaaminen, tietojärjestelmätoimitus, projektitoimitusmalli, ohjelmistotuotanto, systeemyö, ohjel-

mistoteknisen käyttöympäristöt sekä johtamisen kolmio. Sähköinen toimintajärjestelmä oli tutkimuksen aikana kohdeyrityksen työntekijöiden saatavilla.

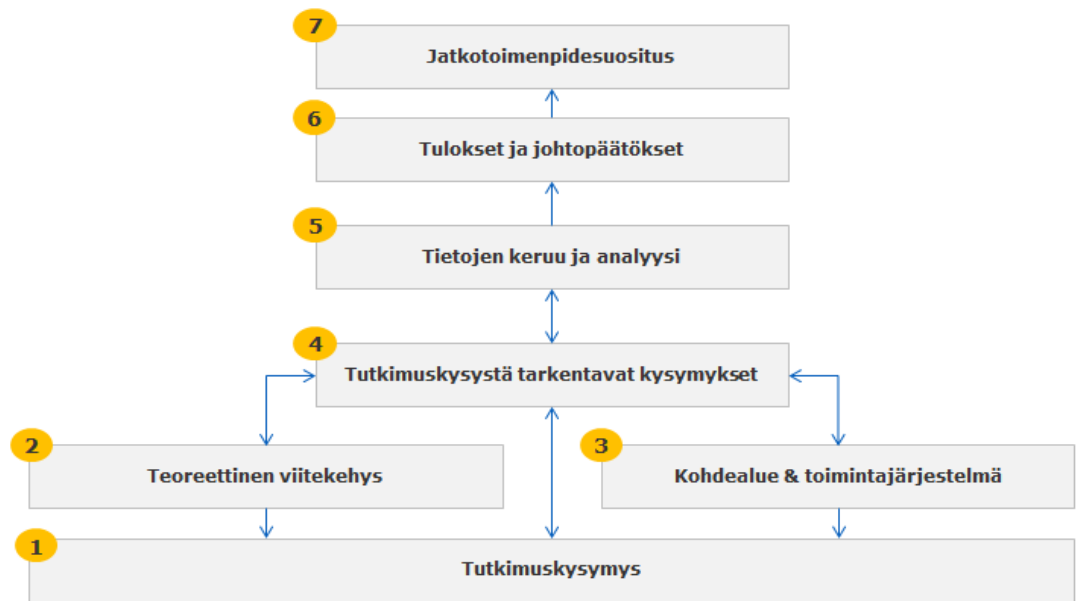
Keskityin tutkimuksessa edellä mainittuihin näkökulmiin huomioiden keskeisen teorian sekä tein havaintoja näkökulmien asioiden leikkauskohdista ”harmaalta vyöhykkeeltä” käsin. Tutkimuksessa en käsittellyt rajauksen mukaisesti liiketoimintaa, liiketoiminnan johtamista, prosesseja, projektityötä tai muutoshallintaa.

1.3 Tutkimusmenetelmä

Laadullisen (kvalitatiivisen) tutkimuksen avulla pyritään ymmärtämään kokonaisvaltaisesti kohteen laatua, ominaisuuksia sekä merkityksiä. Laadullinen tutkimus on mahdollista toteuttaa erilaisilla menetelmillä siten, että menetelmissä korostuvat kohteen taustaan, tarkoitukseen ja merkitykseen liittyvät näkökulmat. Samassa tutkimuksessa on mahdollista käyttää sekä laadullisen sekä määrällisen tutkimuksen suuntauksia. (Jyväskylän Yliopisto 2011.)

Tein tutkimuksen laadullisena tutkimuksena, mihin sisältyi merkittävässä osassa myös kvantitatiivinen kyselytutkimus. Tutkimuksen tarkoituksena oli selvittää ohjelmistometriikan hyödyntämisen merkitystä ohjelmistoliiketoiminnan johtamisen apuvälineenä kohdeyrityksessä. Työlle asetettu lopputulos edellytti riittävän tilan antamista tutkittavaan ilmiöön liittyville subjektiivisille ajatuksille, tunteille sekä kokemuksille. Tutkimuksen tarkoituksena oli myös kuvailla tutkimuskohdetta ilmiönä kohdeyrityksessä sekä tuottaa tältä pohjalta uusia havaintoja liittyen kohdeyrityksen toimintaan. Sähköisen kyselytutkimuksen avulla selvitin tietojärjestelmätoimituksen osalta ohjelmistometriikan nykytilaa kohdeyrityksessä.

Tutkimuksen yksi keskeinen kulmakivi oli, että tutkimuksen sisältö tarkentui teorian käsittelyn sekä kohdeympäristön kautta esille nousevien tekijöiden myötä. Tutkimuksessa käsiteltävistä teorioista muodostui tutkimuksen teoreettinen viitekehys, mitä vasten loin **empiirisen** kyselytutkimuksen sisällön. Käsittelin kerätyn kyselytutkimuksen aineiston **induktiivisen päättelyn** keinoin, jotta sain nostettua esille kohdeyrityksen kannalta merkityksellisiä teemoja sekä havaintoja osaksi työlle määriteltyä lopputulosta.



Kuva 1: Tutkimuksen rakenne

Tutkimuksen rakenne (kuva 1) perustui (1) tutkimuskysymykseen, mihin pohjautuivat (2) teoreettisen viitekehyksen sisältö, rajaus sekä (3) kohdeyrityksen toimintajärjestelmän soveltuvat toimintamallit sekä prosessit. Teoreettisen viitekehyksen ja kohdealueen toimintamallin rajasin (4) ala-tutkimuskysymysten avulla. Näiden tarkentavien kysymysten avulla sekä koostetun viitekehyksen pohjalta laadin kyselytutkimus sekä suoritin (5) tietojen keräämisen sekä analysoinnin. Edellisen vaiheen tulosten pohjalta (6) kokosin tulokset ja havainnot, muodostin johtopäätökset sekä vastasin tutkimuskysymykseen. Tämän perusteella muodostin jatkotoimenpidesuosituksen (7) eli varsinaisen tutkimuksen lopputuloksen.

1.4 Tutkimuksen rakenne

Tutkimus rakentuu osuiksista johdanto, teoria, kohdeyritys, tutkimusmenetelmä, tutkimustulos sekä johtopäätökset ja suositukset. Luvussa yksi kuvaan tutkimuksen taustan, päämäärän, tavoitteet, tarkastelunäkökulmat, rajaukset sekä tutkimustragian.

Luvussa kaksi kuvaan tutkimuksen teoriaosuuden. Sisällöllisesti keskityn teoriaosuudessa ohjelmistometriikan merkityksen lisäksi ohjelmistometriikan osalta ohjelmistoteknisen laadun, mittaamisen, osaamisen, työvälineiden sekä johtamisen näkökulmiin. Luvussa kolme käsittelen kohdeyrityksen toimintamalleja ja

tarkastelen myös ohjelmistometriikan sijoittumista osaksi kohdeyrityksen toimintaa sekä ohjelmistometriikan hyödyntämismahdollisuuksia ennen toimitusta, toimituksen aikana, toimituksen jälkeen.

Luvussa neljä kuvaan tutkimukseen sisältyvän kyselytutkimuksen. Esittelen kyselytutkimuksen osalta tutkimuskontekstin, aineistoon hankinta- ja käsittelymenetelmät. Luvussa viisi esittelen kyselytutkimuksen sisällöllisten teemojen kautta kyselytutkimuksen tutkimustulokset. Tässä luvussa kokoaan myös kyselytutkimuksen lopputulosten osalta tekemäni havainnot sekä esitän ne kyselytutkimuksen teemojen mukaisesti. Luvussa viisi esittelen myös uuden tutkimuksen aikaa syntyneen malli, vastaan tutkimuskysymykseen ja esitän jatkotoimenpidesuosituksen.

Luvussa kuusi vastaan esitän johtopäätökset sekä esitän arvioni tutkimustulosten yleistettävyydestä ja arvioin tutkimuksen validiteettia. Lopuksi esitän jatkotutkimuskohteet.

Tutkimuksen liitteistä löytyvät lähdeluettelo, kutsu kyselytutkimukseen, tutkimuskyselylomake sekä kyselytutkimuksen vastausaineisto. Lisäksi liitteenä ovat kyselytutkimuksen analysoinnissa käyttämäni koosteet sekä kaaviot.

2 OHJELMISTOMETRIKKA TIETOJÄRJESTELMÄN TOIMITUKSESSA

Tässä luvussa esittelen ohjelmistometriikan roolin ja merkityksen ohjelmistokehityksessä, ohjelmiston lähdekoodissa sekä laatukustannuksien näkökulmasta. Lisäksi ohjelmistometriikan osalta sivuan johtamista sekä päätöksentekoa.

2.1 Ohjelmistometriikka ohjelmistokehityksen käytäntönä

”Ohjelmistometriikka (software metrics) on ohjelmiston arviointi mittaluvuin, jotka kuvaavat ohjelmiston eri ominaisuuksia kuten ymmärrettävyyttä, kompleksisuutta, testattavuutta ja dokumentoinnin tasoa” (Pietarinen).

Ohjelmistokehitys (software engineering) on järjestelmällinen, kurinalainen ja mitattavissa oleva lähetysmistapa ohjelmiston kehitykseen ja ylläpitoon (Wikipedia 2012).

Ohjelmistometriikka tuottaa arvokasta tietoa toimitusjohtajalle, IT-päälliköille, projektipäälliköille, ylläpidon sekä tuen johtamisille, laatuvaikuttajille sekä ohjelmistokehittäjille. Ohjelmistometriikan keskeiset hyödyntämisen teemat ovat benchmarking, estimointi, projektihallinta, tuottavuuden kehittäminen, prosessien kehittäminen, hankinnat, palvelutasosopimukset, virheanalyysit sekä teknologiat sekä ympäristöt. Ohjelmistometriikkaan liittyviä vaatimuksia esiintyy useissa kehikoissa (framework) kuten CMMI, COBIT, ITIL, Agile sekä Six Sigma. (QESP 2014.)

Ohjelmistometriikka on osa ohjelmistokehityksen ”state-of-the-practise” käytäntöjä. Terminä ohjelmistometriikka tarkoittaa ihmisille eri asioita vaihdellen projektin kustannusten ja työmäärien mallintamisesta sekä ennustamisesta ohjelmistovirheiden jäljittämiseen tai tietokoneiden suorituskyvyn mallintamiseen. Yritykset hyödyntävät metriikkaa ohjelmistoprojektien, prosessien ja tuotteiden seurantaan, ohjaamiseen sekä ennustamiseen. Metriikan hyödyntäminen edellyttää sen määritelmän sekä tarkoituksen ymmärtämistä, jotta ihmiset voisivat osallistua metriikkaan liittyvien käytäntöjen valintaa, suunnitteluun, toteutukseen, tiedon keräämiseen sekä myös hyödyntämiseen. Metriikan osalta on keskeistä, että sen avulla tuotetaan hyödyllistä tietoa. (Westfall 2005.)

Ohjelmistokehityksen metriikan pääalueet ovat: tuottavuus (productivity), laatu (quality), laajuus (size), tehokkuus (effort) sekä kesto (duration). Tuottavuus on varoen käytettävä mittari, koska vertailukelpoisuus ohjelmiston laajuuden, kypsyyssytason, menetelmien ja työkalujen välillä on merkitykseltöntä. Lisäksi tuottavuuden mittaaminen ei ole ylipäättään vertailukelpoista esimerkiksi yksilön, projektin tai organisaatioiden välillä. (Gack 2012.)

Ohjelmistometriikka on monimuotoinen ja tulkinnanvarainen konsepti ohjelmistokehityksen alueella. Ohjelmistometriikan historian yli kolmen vuosikymmenen aikana aihealuetta on tutkittu laajasti ja sen pohjalta on syntynyt paljon ohjelmistometriikkaan liittyviä suosituksia ja de facto -standardeja, mutta vain virallisia standardeja. Ohjelmistometriikkaan liittyviä julkaisuja ei ole laajasti hyväksytty tutkijoiden sekä hyödyntäjien keskuudessa. (Abrain 2010, 3-6.)



Kuva 2: Ohjelmistometriikka kokonaisuuden osana (Westfall 2003.)

Kuva 2 esittää mittaamisen kytkeymisen periaatteen osaksi muuta toimintaa. Ohjelmistometriikka perustuu mittaamiseen liittyviin käytäntöihin, joita sovelletaan prosesseihin, tuotteisiin sekä palveluihin. Ohjelmistometriikan hyödyntäminen edellyttää tarvittavien keinojen sekä tekniikoiden soveltamista käytännön työssä prosessien, tuotteiden sekä palveluiden parissa, jotta tehdystä työstä jää ”jälki” tiedon mittautiedon tuottamiseksi. Ohjelmistometriikan mittaustieto jalostetaan suunnittelua sekä johtamista varten, jotta prosesseja, tuotteita sekä palveluita voidaan kehittää paremmaksi. (Westfall 2003.)

Ohjelmistometriikka on esitettävissä kokonaisuutena ja sitä tukevana prosessina, joka jakautuu seuraaviin vaiheisiin: 1) tavoitteet, 2) mittaaminen, 3) analyysi, 4) toimenpide sekä 5) seuranta. Keskeistä on päättää, mitä seurata ja mitä mitata. Ohjelmistometriikka voi auttaa laatukustannustekijöiden minimoimisessa siten, että ohjelmistometriikkaan liittyvien mittareiden avulla voi helpottaa asioiden todentamista sekä mahdollisten ongelmakohtien paikallistamista. Ohjelmistometriikan mittareiden teorioita tai niiden toimintaa ei tarvitse ymmärtää syvällisesti.

Mittareiden merkityksen ymmärtäminen on olennaista, kuten myös mittareiden arvojen merkitys eri tilanteissa. (Gack 2012).

Linda Westdfallin mukaan ohjelmistometriikka ei ratkaise ongelmia, vaan sen tekevät ihmiset. Ohjelmistometriikka voi tuottaa tietoa, mitä voi hyödyntää päätöksien sekä parempien valintojen tekemiseen. Tämä edellyttää mittaustietojen tulkintataittoa. (Westfall 2003.)

Ohjelmistojen mittaaminen on verrattain uusi teknologia. Ohjelmistojen mittaamisen **käyttöönoton haaste organisaation näkökulmasta kiteytyy kysymykseen, mitä ohjelmistometriikan käyttöönottamiseksi tulee tehdä.** Lisäksi organisaatiolla tulee olla tietotaitoa uuden teknologian käyttöönottamiseksi, sietokykyä muutoksen osalta sekä ylipäättänsä voimavaroja ymmärtää uuden teknologian mahdollisuudet ja hyödyt. (Abrain 2010, 7-8.)

Alain Abrainin sekä Linda Westdfallein näkemysten perusteella on mahdollista todeta ohjelmistometriikan olevan apuväline, minkä hyödyntäminen vaatii tekemisen lisäksi myös mittaustiedon tulkinta- ja soveltamistaitoja. Ohjelmistometriikka mahdollistaa ohjelmiston mittaamisen objektiivisesti koskien ohjelmiston teknistä laatua, laajuutta, monimutkaisuutta sekä ylläpidettävyyttä. Päätöksentekemisessä ”musta tuntuu” mittareiden sijaan olisi mahdollista hyödyntää objektiivisia mittauskeinoja sekä jättää enemmän tilaa ja aikaa subjektiiviselle mittareiden tulkinnalle.

Ohjelmistometriikka on laaja ohjelmistotuotannon sekä ohjelmistotekniikan osa-alue, minkä hallinta vaatii perehtymistä, soveltamista käytännössä sekä yhteisen kielen ja ajatuksen hyödyntämisestä. Ohjelmistometriikka on mittaamisen osa-alue, mikä tulee upottaa osaksi muuta arkipäivän tekemistä. Ohjelmistometriikan soveltaminen ja hyödyntäminen tulee miettiä eri roolien näkökulmasta siten, että rooleittain asian merkitys, tehtävät ja vastuut ovat selkeitä sekä ohjeistettu kokonaisuuden näkökulmasta. Muutoin ohjelmistometriikka jää irralliseksi muusta toimintatoiminnasta.

Käytännössä tämä edellyttää, että ohjelmistometriikka tulee upottaa soveltuvalla tavalla osaksi johtamista, prosesseja, toimintamalleja sekä pelisääntöjä. Työvälineiden avulla voidaan tietyissä tehtävissä automatisoida mittaustiedon tuottami-

nen ja kerääminen toistumaan säännöllisesti, jolloin vaadittu henkilötyön määrää on merkittävästi pienempi. Työvälineet eivät yksinään riitä, vaan mittareiden hyödyntäminen vaatii tavoitteet sekä raja-arvot, joiden puitteissa toimitaan sovitulla tavalla. Lisäksi mittareiden tuottamat arvot vaativat osaamista arvojen tulkinnasta tilanteeseen nähden, jotta mittareita olisi mahdollista hyödyntää osana johtamista.

2.2 Ohjelmistometriikka tietojärjestelmän toimituksen johtamisessa

Meredithin mukaan on yleisesti tunnettu, että mitä ei voida mitata, sitä ei voida johtaa. Meredith mukaan useimmissa organisaatioissa ei mitata ohjelmistokehitystä tai -ylläpitoa. Tällöin organisaatiot menettävät mahdollisuuden objektiivisyyteen mittausprosessissa. (Meredith 2012).

Projektien toimittamiseksi on olemassa erilaisia elinkaarimalleja, aikaraameja sekä vaiheistuksia. Projektin vaihe on erillinen muista projektin selvästi erottuva ajanjakso ja se sisältää projektin kannalta merkittäviä tuloksia ja päätöksiä, mitkä toimivat perustana projektin seuraavalle vaiheelle. Projektin toimittamista erilaiset mallit ovat teollisuusala- ja liiketoiminta-alakohtaisia. (Koskelainen ym. 2008, 24.)

Keskeisiä ohjelmistometriikan hyödyntämisen syitä johtamisen apuvälineenä ovat ohjelmistoteknisten laatukustannusten minimointi sekä ohjelmistokehityksen tuottavuuden lisääminen (Gack 2012). Käytännössä ohjelmistometriikan hyödyntämisen haaste muodostuu johtamisen osa-alueista eli tavoitteista, seurannasta, suunnitellusta poikkeamista aiheutuvista toimenpiteistä sekä päätöksenteosta. Näiden osalta muodostuu muutama keskeinen haaste ohjelmistoliiketoiminnan johtamiselle yksittäisten ohjelmistokehitysprojektin läpiviennin kuin myös kokonaisuuden johtamisen osalta:

1. Miten on mahdollista **asettaa etukäteen** mitattavat tavoitteet ohjelmistotekniselle toteutukselle sekä tekniselle laadulle, koska asialla on vaikutuksia muun kokonaisuuden budjetille, aikataululle sekä resurssitarpeelle?
2. Miten on mahdollista ohjelmiston toteutuksen **aikana seurata sekä myös reagoida** riittävän helposti, nopeasti, ajantasaisesti sekä kustannustehokkaasti ohjelmiston toteutuksen tekniseen laatuun liittyen?

3. Miten on mahdollista **todentaa jälkikäteen** ohjelmiston toteutuksen haluttu tekninen laatu riittävän helposti, nopeasti sekä kustannustehokkaasti?

Tyypillisesti johtamisessa on kiinnostuttu kvantitatiivisista malleista. Mittaustiedon pohjalta tehtävät päätökset edellyttävät mittaustietoon liittyvää epävarmuuksien ymmärtämistä. Ohjelmistojen osalta päätöksen tekemisen yhteydessä emme voi pitää selvyyttenä esimerkiksi mittaustiedon tarkkuutta tai toisettavuutta, mikäli mittaustiedot perustuvat epämääräiseen tai väärin sovellettuun mittausmenetelmään. Ohjelmistoihin liittyvän päätöksenteon osalta hyödynnämme esimerkiksi testaamisen lopettamispäätöksen tekemisen yhteydessä useita erilaisia mittaustietoja, kuten virheiden lukumäärä, ohjelmiston laajuus, testitapauksien lukumäärä sekä virheiden lukumäärän vähentyminen tarkastastelujaksolla. (Abrain 2010, 18-19.)

Ohjelmistometriikka on yksi keino muiden keinojen joukossa tuottaa ohjelmistotuotantoon liittyvää mittaustietoa. Pelkkä mittaminen ei paranna laatua, vaan mittamisella tulee olla peruste sekä ajatus mittaustietojen hyödyntämisestä, jotta mittamisen avulla olisi mahdollista tehostaa toimintaa sekä parantaa laatua. Mittaamisella tulee olla tarkoitus sekä ennen kaikkea tapa hyödyntää mittaustuloksia, kuten esimerkiksi katselmoinneissa, laadun arvioinnissa tai suunniteltaessa ohjelmiston korjaus- tai parannustoimenpiteitä. Mittaaminen ja seuranta edellyttävät työhön osallistuvien ihmisten (roolien) osalta osaamista, yhteistä näkemystä, yhteisiä toimintamalleja, prosesseja sekä työvälineitä. Muussa tapauksessa ohjelmistoteknisen toteutuksen laadun seuranta ja todentaminen jäävät irti muusta kokonaisuudesta sekä sen johtamisesta.

Johtamisessa päätöksien tekeminen on keskeissä asemassa. Hyvään päätökseen tarvitaan päämäärätieto, voimavaratieto sekä tilannetieto. Hyvä päätös edellyttää selkeää päämäärä- ja tilannetietoa sekä tietoa käytössä olevista voimavaroista sekä toimintakyvystä. Näiden tietojen perusteella päätöksen tekijän on löydettävä keinot tarkoituksenmukaiselle teolle. (Kamppinen yms. 2002, 109.)

Ohjelmistoliiketoiminnan asiakaskohtaisesti räätälöityjen tietojärjestelmien toimittamisen erityisenä haasteena on ajantasaisen tilannetiedon tuottaminen päätöksenteon tueksi oikea-aikaisesti sekä kustannustehokkaasti vähäisellä työmäärällä.

Perusongelma muodostuu tietojärjestelmän kehittämisen aikana, missä järjestelmän toteutus etenee tilanteen muutuessa jatkuvasti. Tällöin korostuvat erilaisten tehtävien toistaminen eri ajankohtina, jolloin myös kustannukset kasvavat. Päämäärätieto ja voimavaratiedot ovat yleensä hyvin hallittuja, koska nämä perustuvat lopputulokseen sekä käytettävissä oleviin ihmisiin, työvälineisiin sekä resursseihin. Tilannetieto tulosten on osalta yleensä hallittu. Tulosten tuottamiseen liittyvä ”alemmman tason tilannetiedon” tuottaminen ja koostaminen jäävät helposti kertaluonteiseksi tai satunnaiseksi tietojen tuottamisen työläydestä johtuen. Tämä johtaa helposti toimintatapaan, missä ohjelmiston toteutukseen liittyvien asioiden käsittely sekä päätöksen teko tapahtuu huteralta pohjalta - ”musta tuntuu” periaatteella ilman faktatietoa.

2.3 Laatukustannustekijöiden minimointi ohjelmistometriikalla

Ohjelmiston julkaisun jälkeen ongelmien korjaaminen tai epäonnistuminen on merkittävästi kalliimpaa kuin ennen ohjelmiston julkaisua (Carpens 2011, 146-148). Ohjelmistovirheiden korjaamiskustannuksien ero työn alkuvaiheessa verrattuna asiakkaalle jaetusta versiosta saattaa olla 40 - 10.000 kertainen (Pressman 2000, 197).

Laajojen tietojärjestelmätoimituksien kehittämis- sekä käyttökustannuksia ovat suuria. Vastaavasti ohjelmistojen virheiden tai ongelmien korjaaminen aiheuttavat merkittäviä kustannuksia, mikä on pois taloudellisesta tuloksesta. Paremman taloudellisen tuloksen kuin myös asiakaslupauksien lunastamisesta muodostuvat keskeiset syyt huomioida ohjelmistotekninen laatu.

Laatukustannukset voidaan jakaa kolmeen ryhmään: ennaltaehkäiseviin, esille tulevien ongelmien korjaamiseen sekä epäonnistumiseen liittyviin kustannuksiin. Ongelmien ennaltaehkäisy sekä välitön korjaaminen ovat kriittisiä tekijöitä ohjelmistoon liittyvien laatukustannusten minimoimiseksi. (Carpens 2011, 146-148.)

Cary Gackin esitteli ITMPI:n webinaarissa 2012 ajatuksia tekemisen tavasta sekä asioiden parantamisesta: *”No single best way to do things and improve things. Six Sigma, ITIL, CMMI etc. Melt best ideas from multiple sources”*. Gackin mukaan tulisi keskittyä isoon kokonaiskuvaan sekä laatukustannustekijöihin tunnistami-

seen erityisesti sisäisten laatukustannusten tekijöiden sekä arvoa tuottamattomien tehtävien osalta. Gack esittää, että toiminnan lisäarvoa parannetaan erityisesti ennaltaehkäisevän, tekemisen aikana esille tulevien arvoa tuottamattomien sekä uudelleen tekemisen liittyvien laatukustannuksien minimoinnilla. Käytännössä tämä edellyttäisi ohjelmistoon liittyvän mittauksen näkökulmasta soveltuvien raja-arvojen asettamista ennen ohjelmiston toteutusta, seuranta toteutuksen aikana, todentamista toteutuksen jälkeen sekä tarvittaessa välittömien korjaavien toimenpiteiden suorittamista. (Gack 2012.)

Laadun määrittely on tyypillisesti vaikea tehtävä. Ohjelmiston laadun määrittäminen on poikkeuksellisen vaikeaa. Ohjelmiston laadulle voidaan asettaa monia kriteerejä, mutta tärkeää on se, että laadun tulee olla määritelty ennen projektin alkamista. Toiseksi laadun määritelmän sekä laadun tulee kehittyä ajan kuluessa, jolloin laadun tulee olla mitattavissa sekä todistettavissa projektin aikana kuin myös sen päättymisen jälkeen. Kolmanneksi laadun tulee olla joustava, koskea kaikkia projektin työvaiheita, tehtäviä ja lopputuloksia. Neljänneksi laadun tulee olla sovellettavissa myös uusiin teknologioihin (Carpens 2011, 9-11).

Carpensin esittämät laatuksiteerit sopivat yhteen ohjelmistometriikan kanssa. Siinä missä laatu tulee määritellä ennen projektin alkamista, niin vastaavasti ohjelmistometriikan osalta tulee valita soveltuvat mittarit sekä niiden raja-arvot ennen työn tekemistä. Ohjelmistometriikka on tuotettavissa työvälineiden avulla haluttuna ajankohtana ohjelmiston lähdekoodista. Ohjelmiston versionhallinnan soveltamisella laadun todistaminen ohjelmistometriikan avulla on mahdollista kesken olevan ohjelmistoversion kuin myös aikaisempien ohjelmistoversioiden osalta.

Carpen jakaa ohjelmiston laadun 121 laatutekijään, mitkä ovat jaettu seuraaviin osa-alueisiin (Carpen 2011, 11-15.):

- Tekninen tai rakenteellinen laatu (luotettavuus, virheet, virhekorjaus)
- Prosessin laatu (sisältäen laatua tuottavat kehittämisen menetelmät)
- Käytettävyyden laatu (helppo oppiminen, helppokäyttöisyys)
- Palvelun laatu (tukihenkilöstö)
- Esteettinen laatu (käyttäjä tyytyväisyys)
- Standardit (suuri joukko laatutekijöitä kansainvälisistä standardeista)

- Oikeudellinen laatu (valitukset, takuu)

Ohjelmistometriikka sijoittuu Carpenin laatutekijöiden luokittelussa ensisijaisesti teknisen ja rakenteellisen laadun osa-alueelle. Ohjelmistometriikan soveltaminen parantaa myös ohjelmistokehitysprosessin laatua.

Ohjelmistometriikan avulla on mahdollista pyrkiä minimoimaan ohjelmiston lähdekoodin laatukustannuksia esille tulevien ongelmien osalta. Ohjelmistometriikka voidaan ensisijaisesti mieltää laatukustannustekijöiden ennalta ehkäisijäksi, mikäli ohjelmistometriikkaa hyödynnetään ohjelmiston toteutuksen yhteydessä. Muussa tapauksessa ohjelmistometriikkaa voidaan hyödyntää jo tehdyn työn teknisen laadun todentamisen tukena.

Käytännössä tulisi ohjelmistokehittäjällä tulisi olla tiedossa soveltuvat ohjelmistometriikan mittareiden hyväksyttävät raja-arvot jo ennen ohjelmiston toteutustyön aloittamista. Lisäksi ohjelmistokehittäjälle tulisi olla ohjelmiston toteutustyön aikana käytettävissä riittävän yksinkertainen työväline, minkä avulla olisi mahdollista todentaa jatkuvaluontoisesti ohjelmiston toteutuksen olevan hyväksyttävissä raja-arvoissa ennen lopputuloksen luovuttamista eteenpäin. Ohjelmistokehityksen osalta jatkuva valmistuneiden ohjelmisto-osuuksien todentamisen vastuu tulee olla sovitulla roolilla. Tarvittaessa myös kokonaisuuden näkökulmasta mittaukselle asetetut poikkeamat tulee korjata välittömästi niiden havaitsemisen jälkeen tai muutoin perustella sekä kirjata poikkeaman hyväksyntä.

Tietojärjestelmiin liittyvien ohjelmistojen toteuttaminen on virhealtista johtuen lukuisista syistä. Yksi tapa arvioida ohjelmistototeutuksesta syntyvien virheiden lukumäärää on arvioida ohjelmiston toiminnallisuuden laajuutta toimintopistelasennan (function points) kautta suhteessa eri työvaiheisiin.

Ohjelmistojen virheiden tuottamisen sekä poistamiselle Jones Carpers (2008) esittää kaksi mallia virheiden tuottamisen sekä virheiden esille tulemisesta sekä poistamisesta:

A. DEFECTS INSERTION PER FUNCTION POINT

- | | |
|----------------|-------------|
| • Requirements | 0.75 - 0.84 |
| • Design | 1.50 - 1.69 |
| • Code | 1.75 - 1.97 |

- Documents 0.50
- Bad Fixes 0.50

B. APPRAISAL METHOD % removed

- Unit Test 25 %
- Function Test 30 %
- Integration Test 30%
- System Test 35%
- Acceptance Test 25 %
- Inspections 60-90 %

Ohjelmistometriikka osuu ensisijaisesti ohjelmiston toteutukseen (Code) sekä virheiden poistamisessa yksikkötestaukseen (UnitTest). Monimutkainen tai laajaohjelmisto sisältää enemmän virheitä kuin pieni ja yksinkertainen ohjelmisto. Lisäksi monimutkainen toteutus tai laaja ohjelmisto edellyttävät huomattavan suuria testauksen työmääriä sekä samalla myös ohjelmiston ylläpito vaikeutuu. Liiketoiminnallisesta sekä taloudellisesta näkökulmasta voi päätyä pohtimaan esimerkiksi kysymystä, kuinka paljon prosentin tehostaminen ohjelmiston toteutusvaiheessa vaikuttaa jäljellä olevaan työkokonaisuuden kerrannaisuuksien muodossa.

Ohjelmistometriikan osalta keskeinen ajatus on se, että ohjelmistometriikan hyödyntämisen avulla ei estetä ongelmia, mutta sen avulla voidaan minimoida sekä ennalta ehkäistä ohjelmistoteknisiä ongelmia teknisen toteutuksen monimutkaisuuden osalta. Ohjelmistometriikan sekä työvälineiden avulla on toteuttajan, testaajan, ohjelmistoarkkitehdin tai muun laatuvaastaavan on mahdollista paikallistaa esimerkiksi liian suuren syklomaattiset kompleksisuudet. Nämä ovat ohjelmiston osalta monimutkaiset, virhealttiit, hankalasti korjattavat sekä ylläpidettävät osuudet, joiden työstäminen vaatii paljon aikaa, resursseja sekä aiheuttavat merkittäviä kustannuksia.

2.4 Ohjelmiston lähdekoodin teknisen laadun merkitys

Ohjelmistometriikan avulla mitataan ohjelmiston lähdekoodia. Asiakaskohtaisesti räätälöidyn tietojärjestelmän ja siihen kuuluvien ohjelmistojen tekninen toteutus vaatii ohjelmointityötä, jolloin ohjelmistokehittäjä kirjoittaa ohjelmiston lähdekoodia käyttäen valittua ohjelmointikieltä. Lähdekoodin avulla ilmaistaan ohjelmiston suoritus ja toiminnallisuus ohjelmiston määrittelyiden sekä teknisen suun-

nittelun perusteella. Lähdekoodin kirjoittaminen on virhealtista johtuen teknisistä virheistä tai ajatusvirheistä. Suuret virheet johtuvat haasteista ohjelmiston vaatimuksissa, määrittelyissä tai suunnittelussa, jolloin lähdekoodia kirjoitetaan vääränlaiseen tarpeeseen.

McCabe cyclomatic complexity on ohjelmistolaatuun liittyvä metriikka, mikä ilmaisee ohjelmiston rakenteen monimutkaisuuden. Mitä suurempi on arvo, sitä monimutkaisempi on sovellus ymmärtää ja sitä suuremmalla todennäköisyydellä se sisältää virheitä. (Chambers & Associates Pty Ltd 2014.)

Spagettikoodi on klassinen ja kuuluisin käsite ohjelmointitavasta vailla selkeää rakennetta. Tämän kaltaiset ohjelmistot ovat erittäin vaikeita ylläpitää, laajentaa eikä ohjelmistototeutuksessa ole käytännössä mahdollisuutta uusiokäytölle. Spagettikoodista johtuvat ongelmat vaikeuttavat ohjelmiston kehittämistä, mistä ongelmat siirtyvät myös ohjelmiston ylläpidon puolelle. Pahimmillaan ohjelmiston lähdekoodin ylläpito vaatii enemmän kustannuksia kuin ohjelmiston toteuttaminen uudestaan. (Brown, Malveau, McCormick III, Mobbray 1998, 119 - 121.)

Esimerkkinä yksi ohjelmistometriikan soveltamista ohjelmistokehittäjien keskuudessa on paikallistaa lähdekoodista esimerkiksi mahdolliset spagettikoodia sisältävät osuudet. Käytännössä ohjelmistometriikan soveltamista on varsin yksinkertainen. Sovelluskehittäjän ei tarvitse tuntea sisällöllisesti monimutkaista teoriaa sekä mittausmallia. Käytännön tekemisen näkökulmasta riittää, että sovelluskehittäjä tuntee kyseisen keinon, siihen liittyvät raja-arvot sekä toimintamallin, mikäli raja-arvot eivät toteudu. Tässä yhteydessä mielenkiintoinen jatkoajatus on, että miten tämänkaltaisen toiminta ja millaisessa muodossa saadaan nostettua hyödynnettäväksi yksilötasolta työryhmätasolle, projektitasolle, toimituksen tai liiketoiminnan johtamisen tasoille.

Ohjelmiston nopea tekninen toteutus eli koodin kirjoittaminen edellyttää myös ohjelmiston koodin helppoa luettavuutta. Uuden ohjelmistokoodin kirjoittamisen osalta sovelluskehittäjän tulee lukea jatkuvasti aiemmin kirjoitettua koodia. Ohjelmiston koodin lukemisen ja kirjoittamisen vaadittavan ajan suhde saattaa olla yli 10:1, mitä asiana ei voi ohittaa. Tästä syystä ohjelmiston koodin tulee olla selkeää. Ohjelmistojen toteutukseen tarkoitetut ohjelmointikielet eivät tee ohjelmis-

toista yksinkertaisia, vaan hyvä ohjelmoija saa toteutuksen näyttämään yksinkertaiselta. (Martin 2009, 12-14.)

Robert Martin mukaan ohjelmistokoodin hyvä luettavuus on tärkeää. Määritelmä hyvälle ohjelmistokoodille on subjektiivinen asia, mihin vaikuttaa suurelta osin ohjelmointikieli, tekninen toteutustapa sekä ennen kaikkea ohjelmistokoodia arvioivan tai lukevan henkilön osaaminen, näkemykset sekä koulukunta. Kirjallisuudesta sekä Internetistä löytyy loputtomasti määritelmiä hyvälle lähdekoodille.

Hyvä ohjelmiston tekninen toteutus koodin osalta ei kuitenkaan riitä. Ohjelmiston koodi tulee pitää puhtaana ajan kanssa, koska ohjelmistokoodilla on tapana ryvettyä sekä heikentyä ajan kuluessa. Tästä syystä ohjelmistokehityksessä tulee jatkuvasti olla joku aktiivisessa roolissa ohjelmiston koodin ryvettymisen sekä heikentymisen estämiseksi. (Martin 2009, 14.)

Jokainen aluksi nopeasti edennyt ohjelmistoprojektin tekninen toteutus hidastuu sekä tuottavuus heikkenee ajan kuluessa. Vaikeaselkoinen ohjelmistokoodi saa ohjelmiston toteutuksen sekä ylläpidon tuottavuuden laskemaan ohjelmiston toteutukseen tehtävien muutoksien takia. Jokainen muutos saa ohjelmiston jakaantumaan useampaan pienempään osaan. Lisäksi jokainen lisäys tai muutos ohjelmistoon vaatii ”teknisiä kikkoja”, joista tulee tietää. Tästä syystä ajan kanssa ohjelmistosta tulee suuri ja sekava. Näiden muutoksia takia ohjelmiston toteutuksesta vastavan tiimin tuottavuus laskee ajan kanssa kohti nollaa, koska aikataulu- sekä tuottavuuspaineiden takia ohjelmistokehitystiimissä olevat ryhtyvät tekemään enemmän ja enemmän ”teknisiä kikkoja”, jolloin tiimin tuottavuus laskee entisestään. (Martin 2009, 4.)

Oman kokemukseni mukaan ohjelmistoprojekteissa keskitytään herkästi tekemään juuri se, mikä on ehdottoman pakko, jotta projektissa päästään eteenpäin. Budjetti sekä aikataulut eivät tyypillisesti sisällä pelivaraa kaupallisista sekä sopimuksellisista syistä. Käytännössä ohjelmistoprojekteissa on jatkuva pakottava tarve edetä ohjelmiston toteutuksen osalta. Tämä johtaa helposti tilanteeseen, missä projektipäällikkö, projektin ohjausryhmä tai projektin ulkopuolelta tapahtuva johtaminen pakottaa projektin etenemään, vaikka projektiryhmä vaatii parantamaan laatua ennen siirtymistä seuraaviin työvaiheisiin. Tällöin projekteissa ryhdytään teke-

mään Robert Martinin mainitsemia ”teknisiä kikkoja” samalla kun myös tietojärjestelmän vaatimukset tai määritykset muuttuvat.

Aikataulu- ja budjettipaineet ovat ymmärrettäviä ja useasti myös asioita, joiden kanssa on vain yksinkertaisesti pakko toimia niiden edellyttämällä tavalla. Erityisen haaste saattaa löytyä johtamisen puolelta näkemyksistä ja käsityksistä. Esimerkiksi toteutuksessa oleva ohjelmisto saattaa näyttää olevan pääosin valmis, jolloin johtamisen osalta saatetaan olettaa muutoksien tekemisen olevan pieni työ. Näissä tilanteissa pieni työ uppoaa helposti osaksi suurempaa kokonaisuutta eikä ohjelmistoprojektissa välttämättä käynnistetä muutoshallintaa aikataulun, kustannuksien tai sisällön osalta. Lisäksi päätettäessä pienistä ohjelmistomuutoksista saattaa päätöksenteon pohjalla olla käytettävissä muutamien henkilöiden tekemä ”musta tuntuu” tyyppinen arvio, jolloin olemassa olevan ohjelmistoratkaisun laajuus, rakenne sekä tekninen monimutkaisuus saattaa jäädä vaille huomiota. Tältä pohjalta periaatteella pinoon pinon päälle lukuisten pienien tai suurempien muutoksien myötä ohjelmiston tekninen toteutus alkaa ryvettyä sekä tuottavuus laskea Robert Martinin kuvaamalla tavalla.

Edellä kuvatuista syistä ohjelmiston teknisen ryvettymisen estämiseksi on kohdeyrityksessä olemassa perusteltu tarve toteutuksen laajuuden, rakenteen sekä monimutkaisuuden objektiiviseksi mittaamiseksi. Tämän mittaamiseen tulee tapahtua säännöllisesti, nopeasti, vaivattomasti sekä kustannustehokkaasti, jotta mittaamista olisi ylipäättänsä mahdollista tehdä. Toiseksi mittaamisen tulisi tuottaa riittävän yksinkertainen indikaattori jokaisesta mitattavasta asiasta, jotta näitä mitareita olisi mahdollista hyödyntää operatiivisessa johtamisessa sekä päätösten tekemisessä.

2.5 Ohjelmistokehityksen menetelmät, ympäristöt ja työvälineet

Tietojärjestelmien sekä ohjelmistojen kehittäminen, testaaminen, kouluttaminen sekä tuotantokäyttö tapahtuvat ohjelmistoteknisessä käyttöympäristössä. Ohjelmiston näkökulmasta yhteen tietojärjestelmään liittyy useita käyttöympäristöjä tukemaan erilaisten työkokonaisuuksien tekemistä.

Prosessina asiakastarpeen täyttävien tietojärjestelmien tuottaminen on erittäin haastava suuren henkilöpanoksen vaativa työkokonaisuus, mihin liittyy satoja tai tuhansia erilaisia tehtäviä. Lisäksi tietojärjestelmien laajuus sisällöllisesti kuin myös toiminnallisesti ovat tyypillisesti kokoluokkaa, että niiden käsittelemiseksi sekä hallitsemiseksi tarvitaan ohjelmistotuotannon menetelmiä tukemaan ohjelmistoprojektin läpivientiä.

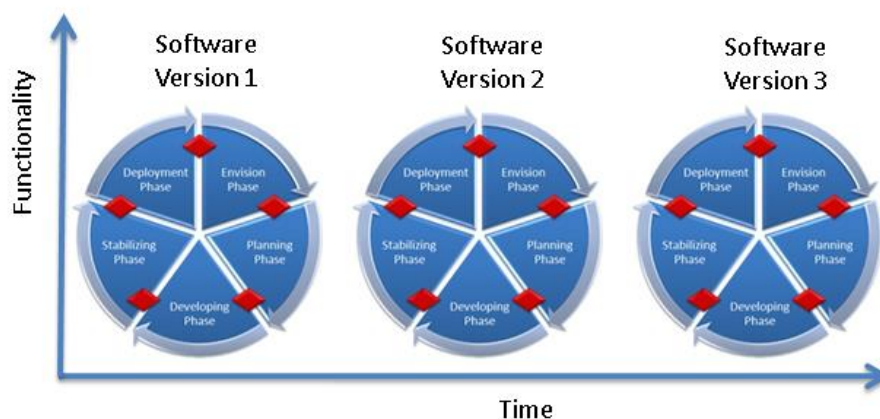
Perinteisissä menetelmissä on pyritty hallitsemaan muutoksia ohjelmistokehityksen aikana, vaikka nämä menetelmät ovat pikemminkin estäneet muutoksien tekemisen. Yksi varma asia on, että asiakkaiden tarpeet ja vaatimukset muuttuvat projektin edetessä. Tästä syystä muutoksia sallivat nykyaikaiset menetelmät auttavat edistämään projektin etenemistä samanaikaisesti muuttuvien vaatimuksien kanssa. Yksi vartenotettava vaihtoehto tämän kaltaiseen työskentelyyn on evolutionaarinen kehittämisen tapa. Evolutionäärinen mallin hyödyt ovat hukkatyön minimointi, merkittävien uudelleen toteutuksien välttäminen, järjestelmän toimivuuden taso on aina tiedossa, ratkaisu on suunniteltu aina parhaalla mahdollisella laatutasolla sekä kokonaistymäärän minimointi. Evolutionäärinen mallin haitat ovat työtavan muutoksien kulttuuriesteet, työvälineiden ja menetelmätuen vajavaisuus sekä jatkuva kehittyminen, oppimiskäyrä, uusien työtapojen oppiminen ja omaksuminen vaativat aikaa. (Amber, Saladage 2008, 22-23.)

Ohjelmistokehityksen yksi keskeinen teema on ollut viimeisen vuosikymmenen ajan ohjelmistojen ketterät työtavat, kuten esimerkiksi SCRUM. Tässä lähestymistavassa arvostetaan 1) yksilöiden välistä vuorovaikusta prosessien ja työvälineiden sijaan, 2) toimiva sovellusta kaiken kattavan dokumentaation sijaan, 3) yhteistyötä asiakkaan kanssa sopimusneuvottelun sijaan sekä 4) muutokseen vastaamista suunnitelman noudattamiseen sijaan (Agile Alliance 2001).

Microsoftin Solutions Framework (MSF) on menetelmäkehikko, minkä avulla on tarkoitus toimittaa ohjelmistojen iteratiivisesti ohjelmistojulkaisuja nopeasti muuttuvassa ympäristössä vähentäen samalla riskejä. Periaatteena on tuottaa aina ohjelmistosta toimiva versio. MSF olettaa, että projektin ajan tulee muutoksia rajoitukseen, teknologiaa sekä henkilöihin. Malliin sisältyvät selkeät hyväksyntäprosessit, joissa hyväksytään päättyvän vaiheen tulokset ennen seuraavan vaiheen aloittamista. (Resnick, Bjork, Maza 2001, 15-16.) MSF:n sisältämä prosessimalli so-

veltuu muassa perinteisten sovelluksien, yrityssovellusten sekä verkkosovelluksien tuottamiseen (Microsoft 2003, 2-5).

Microsoftin MSF Process Model -mallin mukaan laajemman ohjelmiston kehittäminen voidaan jakaa useampaan iteraatioon, joista jokainen iteraatio tuottaa uuden ohjelmistoversion ennen valmista ja julkaistavaa ohjelmistoversiota. Prosessimalliin sisältyy jokaisen ohjelmistoversion kehittäminen, testaaminen sekä julkaisu. Kuvassa 3 on esitetty Microsoft Solution Frameworkin perusajatuksien mukaisesti yksinkertaistetuna ohjelmiston toiminnallisuuden rakentuminen vaiheittain eri versioiden muodossa. Tässä ohjelmistokehityksen lähestymistavassa ohjelmistoversio on aina testattu sekä toimiva kyseiseen ohjelmistoversion julkaisuun määritellyllä toiminnallisuudella. Työstettävässä iteraatiossa laajennetaan edellisen iteraation tuottamaa ohjelmistoversiota. (Microsoft 2003, 12-14.)

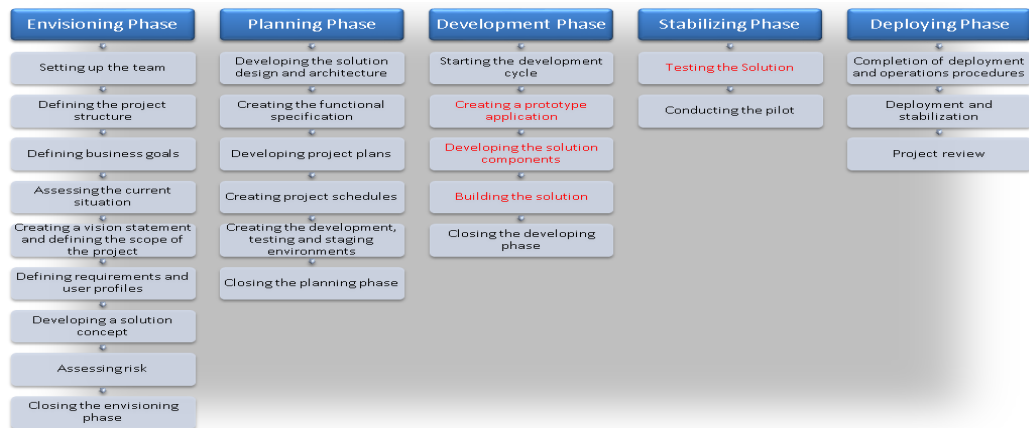


Kuva 3: Ohjelmistoversioiden julkaisu MSF-prosessimallissa (Microsoft 2003, 14.)

Mielestäni ohjelmistometriikan hyödyntäminen sopii erinomaisesti yhteen iteroivien lähestymistapojen kanssa. Ohjelmiston rakentamisen iteroivalla tavalla tarjoaa luonnostaan keinot asettaa tavoitteita, mitata sekä todentaa niitä. Lisäksi todetun poikkeaman korjaaminen on mahdollista tehdä myös seuraavan iteraation aikana, mikäli korjauksen tekeminen ei ole mahdollista tehdä välittömästi.

Microsoft MSF Process Model sisältää vaiheistuksen, missä kussakin vaiheessa on päätason tehtäviä. Kunkin päätason tehtävän alle sisältyy joukko muita tehtäviä, jolloin prosessimalli sisältää lukumäärällisesti erittäin paljon tehtäviä. Ohjel-

mistokehitykseen sekä ohjelmiston testaamiseen liittyvät päätason tehtävät ovat korostettu kuvassa 4 punaisella tekstillä.

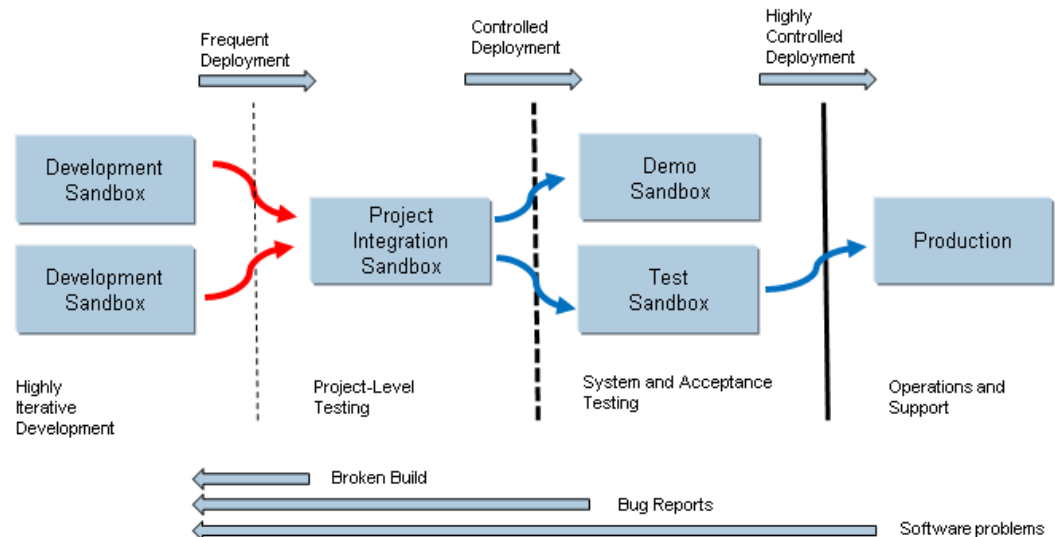


Kuva 4: MSF vaiheistus ja päätason tehtävät (Microsoft 2003, 15-24.)

Ohjelmistometriikan soveltamisen kannalta käytettävän työmenetelmän osalta tulee tiedostaa, mihin tehtäväkokonaisuuksiin ohjelmistometriikan soveltaminen on perusteltua tai ylipäättänsä mahdollista. Ohjelmistometriikka tulee kytkeä soveltuvalle tavalla osaksi muita prosesseja sekä tekemisen malleja, jotta siitä on saatavissa hyötyä. Tästä syystä ohjelmistometriikka tulee huomioida myös työmenetelmissä.

Laajempien ohjelmistojen rinnakkainen kehittäminen, testaaminen sekä käyttö vaativat useita käyttöympäristöjä. Rinnakkaisten käyttöympäristöjen tarkoituksena on mahdollistaa ohjelmiston turvallinen samanaikainen kehittäminen, testaaminen sekä käyttö. Perusajatuksen on, ettei iteratiivinen kehittäminen häiritse samanaikaista kehittämistä eikä testaus häiritse ohjelmiston tuotantokäyttöä. Kuvassa 5 kehittäjän hiekkalaatikko (Development Sandbox) on ohjelmiston teknisen toteutuksen ympäristö, missä sovelluskehittäjä tekee ohjelmistoteknisen toteutustyön sekä ohjelmiston teknisen yksikkötestauksen. Kehittäjältä valmistuneet ohjelmistot osiot tai komponentit siirretään projektin integrointitestiympäristöön (Project Integration Sandbox), missä varmistetaan valmistuneiden ohjelmistojen osien toimivuus osana muuta kokonaisuutta ohjelmistokehittäjien sekä testaajien toimesta. Testajien suorittamien kehitysprojektin sisäisten testien jälkeen suoritetaan ohjelmiston systeemitestit (System Test, Test Sandbox) ohjelmiston toiminnan testaamiseksi. Systeemitestien jälkeen suoritetaan ohjelmiston varsinainen hyväk-

symistestaus (Acceptance Test, Test Sandbox), minkä tarkoituksena on varmistaa ohjelmistoversion kelpoisuus tuotantokäyttöön (Production). (Amber, Sadalage 2008, 10-11.)



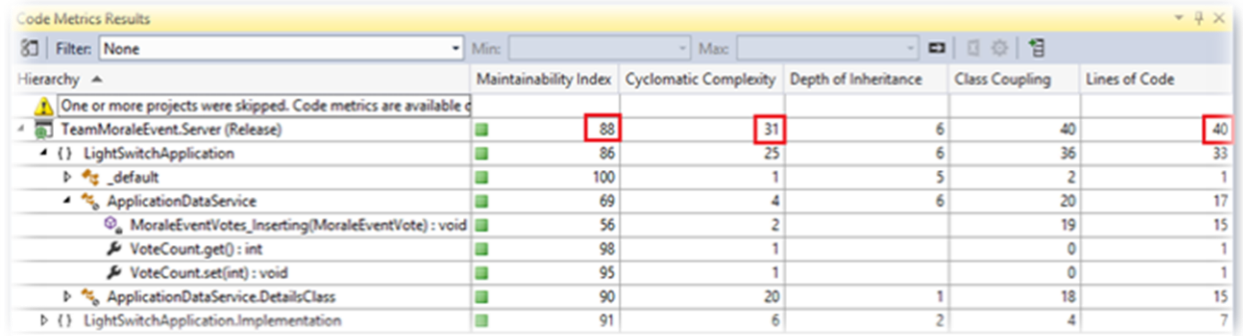
Kuva 5: Ohjelmiston käyttöympäristöt (Ambler, Sadalage 2008, 11.)

Ohjelmistokehitykseen tarkoitettut työvälineet, kuten ohjelmistometriikka, sijoittuvat kehitysympäristöön (Development Sandbox) sekä usein myös integraatiotestiympäristöön (Integration Sandbox). Ohjelmistokehitykseen liittyvät työvälineet eivät yleensä sisälly tuotanto-, koulutus-, testi- tai demoympäristöihin.

Kohdeyrityksessä on ohjelmistotuotannossa käytössä suurehko määrä ohjelmistotekologioita sekä näitä tukevia ohjelmistokehittäjille suunnattuja ohjelmistometriikkaan tukevia työvälineitä. Tutkimuksessa en käsittele työvälineitä, mutta sivun kahta kohdeyrityksessä käytössä olevaa ohjelmistometriikkaa tukevaa työvälinettä mielikuvan luomiseksi.

Microsoft Visual Studio 2013 on ohjelmistokehittäjän työväline, minkä avulla ohjelmistokehittäjä tekee ohjelmiston teknisen toteutuksen. Työväline sisältää seuraavat ohjelmistometriikan mittarit, joiden avulla tunnistaa mahdollisia riskejä, ymmärtää ohjelmistoprojektin nykytilaa sekä seurata ohjelmistokehityksen edistymistä. Visual Studio sisältää seuraavat mittarit: **Maintainability Index** (ohjelmistokoodin ylläpidettävyys), **Cyclomatic Complexity** (rakenteellinen monimutkaisuus), **Depth of Inheritance** (luokkahierarkian syvyys), **Class Coupling**

(riippuvuudet muiden ohjelmisto-osuuksien kanssa) ja **Lines of Code** (lähdekoodin rivimäärä).(MSDN 2014.)



	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
TeamMoraleEvent.Server (Release)	88	31	6	40	40
LightSwitchApplication	86	25	6	36	33
_default	100	1	5	2	1
ApplicationDataService	69	4	6	20	17
MoraleEventVotes_Inserting(MoraleEventVote) : void	56	2		19	15
VoteCount.get() : int	98	1		0	1
VoteCount.set(int) : void	95	1		0	1
ApplicationDataService.DetailsClass	90	20	1	18	15
LightSwitchApplication.Implementation	91	6	2	4	7

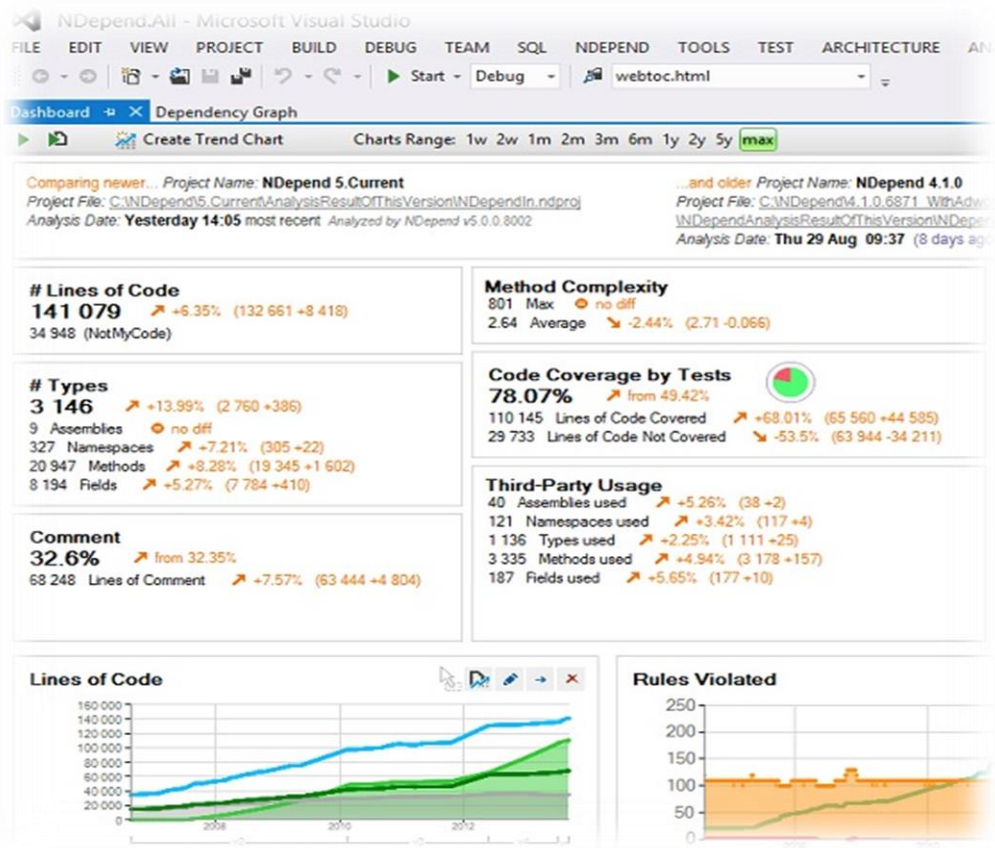
Kuva 6: Visual Studio2013 Code Metrics Output View (Microsoft 2014.)

Visual Studio 2013 mittaa ohjelmistometriikan lähdekoodin perusteella. Kuvassa 6 on esitetty työvälineen tapa esittää mittareihin liittyvät liikennevalot sekä numeerisina arvoina. Yksinkertaistettuna perusajatus on, että mittareiden tulee näyttää vihreää tai muutoin ohjelmiston tekninen toteutus tulee korjata. Visual Studio 2013 on sovelluskehittäjän työväline, minkä avulla sovelluskehittäjä voi seurata sekä todentaa ainakin oman työnsä tuloksia.

Visual Team Foundation Server (TFS) on työväline, mikä on tarkoitettu ohjelmiston elinkaaren hallinnan sekä ohjelmistokehitysryhmän työn tukemiseksi. TFS sisältää ohjelmiston lähdekoodin versionhallintaan tarvittavat ominaisuudet. Lisäksi TFS:n sisältää joukon valmiita sääntöjä hyödynnettäväksi, mitkä tuotetun lähdekoodin tulee läpäistä ennen kuin lähdekoodin tallentaminen versionhallintaan on sallittu. TFS mahdollistaa myös omien tarkastussääntöjen luomisen sekä höydyntämisen. (Blankenship, Woodward, Holliday, Keller 2011, 156-161.)

TFS:n soveltaminen yhdessä esimerkiksi Visual Studion ohjelmistometriikan osalta yksi vaivaton tapa aloittaa ohjelmistometriikan soveltaminen ohjelmistoprojektissa.

NDepend 5 on ohjelmistokehittäjille, testaajille sekä ohjelmiston teknisestä laadusta vastaaville tarkoitettu teknisen laadun seurantaan tarkoitettu kaupallinen työväline. NDepend tarjoaa 82 mittaria mitata lähdekoodin teknistä laatua. Kuvassa 7 on NDepend-työväline, mikä muun muassa suorittaa ohjelmiston mittauksen, tuottaa mittauksesta raportteja sekä yhteenvetoja.



Kuva 7: NDepend 5 Dashboard (Smacchia 2014.)

Ohjelmiston toteutuksen sekä teknisen laadun osalta on tärkeää, että sovelluskehittäjä seuraa päivittäin oman työn jälkeä, jotta mahdolliset monimutkaisuudet tulisi havaittua sekä korjattua mahdollisimman nopeasti ennen testaamista. Tämän kaltaisen toiminnan käynnistäminen vaatii työn tekemisen ohjaamiselta sekä johtamiselta myös oman osuuteensa. Projektipäällikön, laatuvaastaavan tai esimiehen tulee osata vaatia näitä toimenpiteitä suoritettavaksi. Tämän perusteella syntyy osaamis- ja koulutustarve sovelluskehittäjien kuin myös muiden roolien osalta. Projektiryhmälle, työstä vastaaville sekä esimiehille tulisi järjestää ohjelmistometriikkaan liittyvää koulutusta sisällön kuin myös työvälineiden osalta unohtamatta tuottaa ja dokumentoida roolikohtaisia ohjeita.

2.6 Ohjelmistometriikan hyödyntäminen

Yleinen kasku ”Sitä saa, mitä mittaa” toimii myös ohjelmistoliiketoiminnassa. Tästä syystä johtamisen tueksi tarvitaan mittareita, mitkä indikoivat mitattavan asian tilannetta. Johtamisessa pelkät mittarit eivät yksinään riitä. Johtaminen edel-

lyttää tavoitteiden asettamista, seuranta sekä toteuman mittaamista, jotta poikkeamien osalta on mahdollista suorittaa tarvittavia korjaavia toimenpiteitä.

Tutkimuksen rajauksen mukaisesti tarkastelen kohdealuetta asiakaskohtaisen räätälöidyn tietojärjestelmän toimitusta projektina. Tästä kontekstista tarkasteltuna ohjelmistoliiketoiminnassa ja sen johtamisessa korostuvat kilpailu, toimialaosaaaminen, asiaskkaan tunteminen, asioiden soveltaminen käytäntöön, myynti, harvoin tapahtuvat kaupat, sopimus, kilpailu, riskienhallinta, talousasiat, projektitoiminta, tietotekninen osaaminen sekä toimituksen sisällön, resurssien ja osaamisen hallinta. Näitä asioita johdetaan toiminnan kaikilla tasoilla liiketoiminnan eri roolien sekä vastuualueiden toimesta niin asiakkuuksien, myynnin, tarjonnon, henkilöstön, talouden, hallinnon, tuotannon, kehittämisen sekä töiden osalta.

Jukka Ala-Mutkan mukaan strategiamalli on johtamisen kannalta punainen lanka, kehittyvä juoni, yksinkertaistettu kuvaus, systeemin luonne, johdonmukaisuus, kokonaisvaltaisuus, ympäristöönsä vaikuttava sekä reagoiva sekä subjektiivinen kuvaus tekijöidensä mielipiteistä, uskomuksista sekä ajattelumallista. Johtamisessa strategia rakentuu elementeistä, joihin liittyvät mittari, indikaattori, sääntö, arvo, normi, tavoite sekä tehtävä. (Ala-Mutka 2008, 126-127.) Tietojärjestelmän toimituksen johtamisessa tarvitaan myös johtamista sekä johtamisen tueksi mittareita, indikaattoreita, sääntöjä sekä tavoitteita. Ohjelmistometriikan osalta tulisi valita soveltuva mittarit sekä indikaattorit, mitkä tulisi liittää osaksi johtamista.

Jukka Ala-Mutka esittää strategian näkökulmasta iteratiivisen johtamismallin, missä toteutustyyppisen iteraation kesto on kuukaudesta kolmeen kuukauteen. Jokaisen iteraation löydökset liitetään johtamismalliin, jolloin samalla myös näkemykset kehittyvät jatkuvasti iteraatioiden myötä. Lyhyiden iteraatioiden avulla kehittämisestä tulee nopeatemposta että jatkuvaa. Laaja osallistuminen takaa toteutumisen eikä johtamisen strategiamallin jalkauttamista tarvita enää. (Ala-Mutka 2008, 120.) Iteroivassa ohjelmistokehityksessä tarvitaan vastaavasti johtamista lyhyissä sykleissä. Ohjelmistometriikan soveltaminen sopii hyvin tietojärjestelmän toimituksen johtamiseen, koska ohjelmiston tekniseen laadun tai poikkeamien osalta on mahdollista tehdä päätöksiä pienempien kokonaisuuksien osalta, jolloin hukkatyön määrä korjaavien toimenpiteiden osalta pienentyy.

Johtamisen osalta Ala-Mutka (2008) jakaa strategiamallinnuksen neljään tasoon:

- 1. strategiataso (käsitteellinen, periaatteellinen, strategialogiikat)*
- 2. liiketoimintataso (konkreettinen elementtien yhdistelmä)*
- 3. toimintataso (käytännönläheinen, selkeä ohje toiminnalle)*
- 4. henkilötaso (toimijan näkökulma)*

Johtamista sekä sitä tukevia mittareita tarvitaan myös ohjelmistoliiketoiminnan eri tasoilla, joihin soveltuvat Jukka Ala-Mutkan esittämät tasot. Tutkimuksen rajauksen näkökulmasta ohjelmistometriikka sijoittuu Ala-Mutkan mallissa toimintatasoon sekä henkilötasolle. Ohjelmistoliiketoiminnan näkökulmasta ohjelmistometriikkaa tulisi kuitenkin tarkastella kaikkilla Ala-Mutkan esittämillä tasoilla.

Liiketoiminnassa taloudellisten tavoitteiden asettaminen ja saavuttamisen varmistaminen jatkuvan seurannan ja mittaamisen avulla on yrityksissä osa normaalia arkipäivää. Tästä syystä yrityksissä suunniteltuun taloudelliseen tulokseen liittyvät mahdolliset poikkeamat osataan ennakoida ajoissa sekä pyritään myös korjaamaan suunnitelmallisesti. Vastaava toimintamalli tulee lisätä ohjelmistotuotannon eri osa-alueiden seurannan, mittaamisen sekä johtamisen osalta toiminnan eri tasoille.

Ohjelmistoliiketoiminnan johtaminen sisältää suurelta osin samoja elementtejä kuin minkä tahansa muun liiketoiminnan johtaminen. Kohdeyrityksessä päämäärät, tavoitteet, vastualueet, tehtäväksiannot, seuranta, valvonta sekä kehittäminen ovat osa normaalia ohjelmistoliiketoiminnan johtamisen arkipäivää. Ohjelmistoliiketoiminta sisältää kuitenkin omat toimialan erityispiirteensä, mitkä tulee huomioida mittaamiseen osalta.

Ohjelmistometriikan hyödyntäminen edellyttää mittaustiedoston keräämistä. Mittaustiedon keräämisen osalta keskeistä on mittaustiedon keräämisen aloittaminen muutaman projektin osalta vain muutamilla laskettavissa olevilla tiedoilla, kuten esimerkiksi lähdekoorin rivimäärä (Line of Code), Effort (staff months), Time (calendar months) sekä Defects (virheluokittelun perusteella). Liian suuren mittaustiedon keräämisen määrällä on mahdollista päätyä helposti tilanteeseen, missä kerätystä mittaustiedosta tulee merkityksetöntä. Mittaustiedoston osalta on syytä ymmärtää, mitä tietoa on keräämässä, koska kerätyn mittaustiedoston perusteella

voidaan luoda kalibroitu arviomalli tulevaa hyödyntämistä varten. (McConnell 2006, 95-96.)

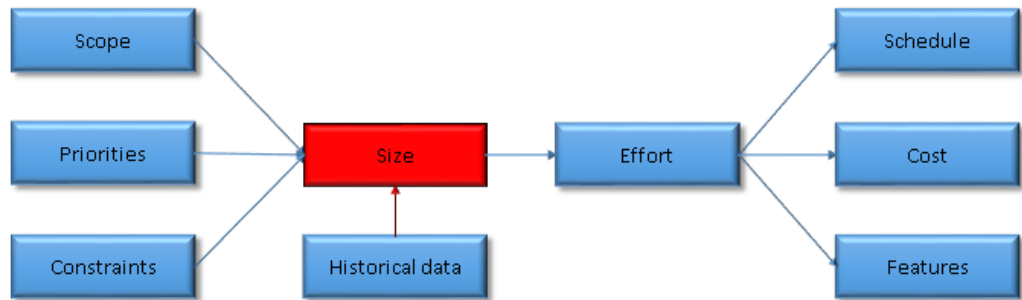
Ohjelmistokehityksen johtamisen näkökulmasta ohjelmistometriikka voi antaa selkeitä keinoja mitata, seurata sekä tarvittaessa korjata tilannetta siinä missä toimitaan kustannus- tai aikataulupoikkeamien osalta. Ohjelmiston teknisen toteutuksen onnistuminen, virheettömyys sekä selkeys ovat ominaan minimoimaan laatukustannustekijöistä aiheutuvia kustannuksia sekä siten vaikuttavat myös keskeisesti työmäärä-, aikataulu- sekä kustannusarvioiden toteutumiseen. Ohjelmistometriikan hyödyntämisen avulla voidaan parantaa kustannus- sekä aikatauluarvioiden toteutumismahdollisuuksia.

Mittaustietoa tulee kerätä säännöllisesti viikon tai kahden välein, jotta kerätystä historiatiedosta on mahdollista luoda profiili projektin etenemisestä tai tilanteen kehittymisestä. Mittaustietojen keräämisen tarkoituksena on saada käsitystä tai luoda malli työmäärien tai muiden arvioiden antamiseksi, esimerkiksi kuinka monta riviä lähdekoodia syntyy keskimäärin yhdestä toimintopisteestä tai kuinka paljon on virheitä keskimäärin 10.000 tai 100.000 koodirivin sovelluksessa. (McConnell 2006, 97-98.)

Ohjelmistometriikan ja siihen liittyvien työvälineiden avulla on mahdollista tuoda automaattisesti esille ajantasainen viimeisin tilanne hyödynnettävien mittareiden näkökulmasta. Käytännössä mittaamisen osalta tulisi selvä malli ohjelmistokehityksen sekä sen johtamisen osalta, mitä mitataan, missä yhteydessä sekä mihin mittareita hyödynnetään. Tällöin erityisesti säännöllisesti toistuvan mittaamisen avulla voidaan ajoissa paikallistaa ohjelmistoteknisen toteutuksen mahdollisia ongelmakohtia, joiden korjaaminen tulee edellyttämään kustannuksia laatukustannustekijöiden osalta esimerkiksi korjaamisen tai uudelleen tekemisen muodossa. Tässä kohdin tulee huomioida virheen tai ongelman korjaamisen kustannuksien kerrannaistuminen tulevissa työvaiheissa.

Ohjelmistotoimituksien osalta ohjelmistometriikan avulla olisi mahdollista määrittellä soveltuvat tekniset laatukriteerit ohjelmistolaadulle sekä hyödyntää näitä kriteerejä osittain myös tarjouksen työmäärä-, aikataulu- sekä kustannusarvioiden tekemisessä. Tällöin esimerkiksi tarjousvalmistelun aikana olisi mahdollista arvi-

oida ohjelmiston lähdekoodin rivimäärä (Line of Codes) ohjelmiston laadun sekä ohjelmiston tekninen monimutkaisuuden (Cyclomatic Complexity) raja-arvot esimerkiksi ylläpidettävän ohjelmiston vastaanottamiseksi. Merkittävä poikkeama määritettyjen ohjelmistometriikan mittareiden arvojen osalta mahdollistaisi johtamisen siinä, missä normaalisesti johdetaan aikatauluun, työmäärään tai kustannuksiin liittyviä poikkeamia.



Kuva 8: Projektin liittyvien arvioiden tekeminen (McConnell 2006, 171-173.)

Ohjelmiston työmäärien, aikataulun sekä kustannusarvioiden osalta ensinmäiseksi tulee arvioida ohjelmiston laajuus (size) sekä tämän jälkeen arvioida tarvittava työmäärä (effort). Lopuksi näiden pohjalta on arvioitavissa aikataulu (schedule), kustannukset (costs) sekä ominaisuudet (features). Ohjelmistoon laajuuden arvioimiseksi tarvitaan lähtötietoina raja-alue (scope), prioriteetit (priority) sekä rajoitteet (constraints). Aiemmin kerättyä historiatietoa hyödynnetään ohjelmiston laajuuden sekä tuottavuuden arvioimiseksi kuvan 8 mukaisesti. (McConnell 2006, 171-173.)

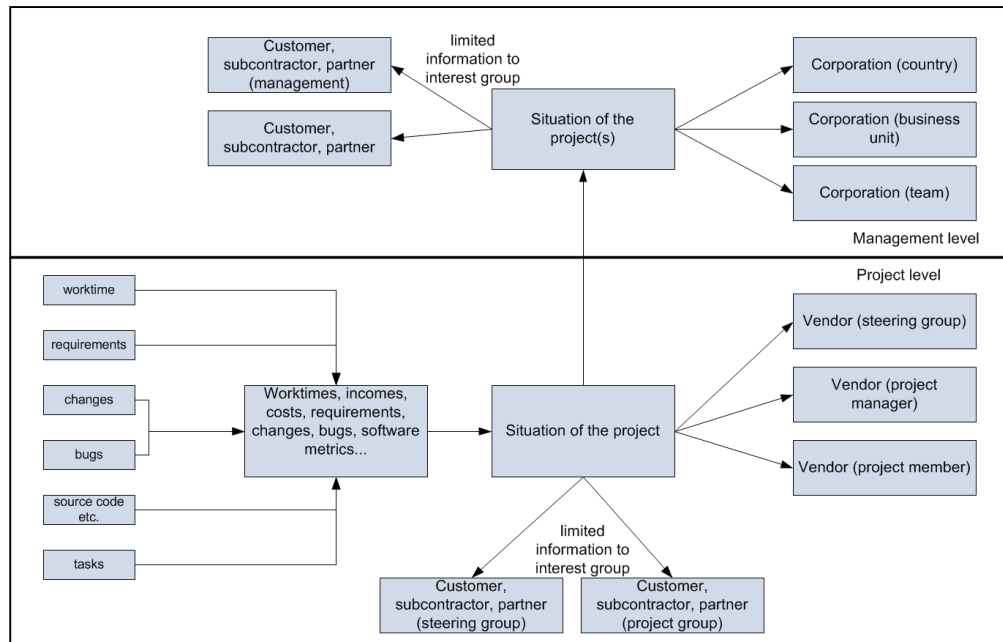
2.7 Ohjelmistometriikan hyödyntäminen toimitusta laajemmin

Energizing Trends in Software Metrics (WETSoM) kansainvälisessä työpajassa yhtenä käsiteltävänä aiheena oli uusien metriikoiden käyttöönotto suurissa organisaatioissa. Metriikan tulisi olla sopivalla tarkkuustasolla sekä johdettu projektitympäristöistä objektiivisesti, jotta metriikka olisi tehokasta sekä käyttökelpoista. Kirjallisuudessa löytyy runsaasti metriikoiden määritelmiä ja kuvauksia, mutta hyvin vähän tutkimuksia sekä keskusteluja siitä, kuinka organisaation tulisi käyttöönottaa sekä käyttää metriikoita. (Sharma, Kaulguld 2013.)

Esimerkiksi Technology & Business Integrators (TBI) kuvaa artikkelissaan pankin tietöjärjestelmän ohjelmistometriikan käyttöönoton ja raportoinnin parantamisen. TBI kehitti pankille metriikan olemassa olevien projektien seurantaan ja valvontaa varten siten. TBI kehitti säännöllisen ohjelmistometriikan raportoinnin ja formaatin operatiiviseen kuin myös strategiseen käyttöön. Operatiiviselle tasolle TBI määritteli perusohjelmistometriikan projektien tilanteen seurantaan sekä analysointia varten. Lisäksi TBI kehitti korkeamman tason ohjelmistometriikan johdolle kommunikointia varten ohjelmistojen laadusta sekä tuottavuustrendeistä päätöksenteon tueksi. (Technology & Business Integrators 2012.)

Jokainen ohjelmistoprojekti sekä niissä tuotettava asiakaskohtainen räätälöity tietöjärjestelmä ovat erilaisia ratkaisukohtaisine erityispiirteineen. Ohjelmistoihin kohdistuvassa mittaamisessa tulee asettaa etukäteen kokonaistilanteeseen nähden soveltuvat raja-arvot. Yhden projektin osalta tietyn mittarin arvo viisi on täydellinen suoritus, kun toisessa projektissa se voi olla täydellinen epäonnistuminen. Yksittäisen projektin osalta tämä tarkoittaisi soveltuvien raja-arvojen valitsemista ennen työn tekemistä, jotta ohjelmistometriikasta olisi projektin ohjaukselle ja mittaamiselle hyötyä. Toiseksi useiden rinnakkaisten projektien kokonaisuuden näkökulmasta tulisi kehittää yksi riittävän selkeä yhteismittallinen mittaus- tai raportointimallin johtamisen tueksi kokonaisuuden kannalta.

Ohjelmistometriikan sekä johtamisen välistä yhteyttä voi tarkastella eri näkökulmista. Kuvassa 9 olen esittänyt jaon projektijohtamisen sekä muun johtamisen välillä sovitettuna kohdeyrityksen tarpeisiin.



Kuva 9: Ohjelmistometriikan keruu, analysointi sekä hyödyntäminen

Kuvassa 9 projektitasolla johtaminen liittyy kiinteästi projektin johtamiseen päämäärän, tavoitteiden, aikataulun, kustannuksien, riskien, laadun, tehtävien, tulok-sien sekä resurssien osalta. Ohjelmistoprojekin osalta ohjelmistometriikka tarjoaa oman näkemyksen projektin tilanteesta, minkä perusteella on mahdollista tehdä tarvittavia toimenpiteitä projektiryhmän sisällä tai asiakkaan, alihankkijoiden sekä kumppaneiden suhteen. Käytännössä projektitasolla tulee projektisuunnitelmaan, valittuun työtapaan sekä menetelmään sisällyttää tarvittavia toimenpiteitä ohjel-mistometriikan osalta. Näiden toimenpiteiden tekeminen edellyttää aikaa, tehtä-vien vastuuttamista sekä kustannuksia. Tärkeä huomioitava asia on, että työn te-kemistä tulee myös johtaa.

Metriikan sopivan soveltamistavan kannalta olisi tärkeää huomioida käytötavat periaatteet. Metriikka tulisi ensinnäkin linkittää tavoitteisiin. Toiseksi mittaami- sessa tulisi keskittyä trendiin yksittäisten arvojen sijaan, koska muutoksen seuran- ta ja sen perusteella tehtävät jatkotoimenpiteet ovat merkittäviä. Kolmanneksi mittaamissykli tulee asettaa lyhyeksi. Lisäksi neljänneksi sovellettu metriikka tulee vaihtaa toiseksi, kun metriikka ei osoita enää muutosta. (Kua 2013.)

Kuvassa 9 projektin ulkopuolella ohjelmistometriikkaa voidaan hyödyntää projek- tien tilanteen sekä teknisen laadun arvioinnissa. Samalla ohjelmistometriikasta voidaan tuottaa johdannaista tietoa esimerkiksi organisaation kyvykkyydestä tuot-

taa tietynlaisia ratkaisuja tietyissä puitteissa. Tällöin kokonaisuuden näkökulmasta voidaan luoda näkemys mahdollisista tarvittavista kehitystoimenpiteistä asioiden parantamiseksi esimerkiksi koulutuksien tai kehitysprojektin muodossa. Lisäksi ohjelmistometriikasta syntynyttä tietämystä voidaan jalostaa sekä hyödyntää uusien projektien tai hankkeiden myymisessä asiakkaille esimerkiksi tarjousvalmistelussa, työmääräarvoinneissa tai sopimuksissa.

Ohjelmistometriikan osalta pelkkä mittarin numeerinen arvo ei kerro hyödyntämisen syytä tai tarvetta. Ohjelmistometriikka tulisi aina linkittää käyttötarkoitukseen tai perusteluihin, jotta ihmisten olisi helpompi mieltää tavoitearvoille asetetut taustat ja syyt. Patrick Kuan mukaan ohjelmistometriikan tavoite tulisi kuvata esimerkiksi seuraavasti: *”We would like our code to be less complex and easier to change. Therefore we should aim to write short methods (less than 15 lines) with a low cyclomatic complexity (less than 20 is good). We should also aim to have a small handful of parameters (up to four) so that methods remain as focused as possible.”* (Kua 2013.)

Organisaation ja yksilöiden tulee oppia tehdystä työstä tekemisen aikana kuin myös tekemisen jälkeen. Ohjelmistometriikkaa tulisi kerätä, analysoida sekä tallentaa organisaation tietämysjärjestelmää uusien vastaavien ohjelmistojen mitoitamisen sekä arvioinnin tueksi. Tehdyn työn kautta on mahdollista analysoida myös tekemisen tavan muuttumista, jolloin organisaation sisäinen tietämys eläisi enemmän dynaamisesti ajanhengessä eikä kuvastaisi staattisesti mennyttä aikaa. Tällöin ohjelmistometriikka voi tuoda uusia näkökulmia toiminnan kehittämiseen ja indikoida paremmin myös organisaation kyvykkyyttä toimia vastaavanlaisten ratkaisuiden tuottamisessa sekä heijastella myös organisaation osaamiseen liittyvää muutosta.

Sovelluskehittäjät voivat myös oppia ohjelmistometriikan avulla tuottamaan entistä parempaa teknistä laatua, kun heillä on käytettävissä keino mitata omaa työtään. Lisäksi työvälineet, kuten esimerkiksi NDepend, ilmaisevat mittareiden lisäksi mittaustuloksen aiheuttaneen syyn sekä ehdottavat korjaus- tai parannusratkaisuja.

Ohjelmistometriikan hyödyntäminen alihankkijoiden tai xshore (offshoren, nearshoren, onshoren, blended shore) työn seurannassa auttaa saamaan yhteismittallista sekä objektiivisesti tuotettua ohjelmiston teknistä laatua kuvaavaa tietoa. Esimerkiksi alihankkijoita olisi mahdollista velvoittaa jo alihankintasopimuksessa toimimaan tietyllä tavalla ohjelmistometriikan osalta, jotta alihankkijan tuottamat ohjelmisto-osuudet täyttäisivät myös muun ohjelmistokokonaisuuden teknisen laadun kriteerit. Samalla myös alihankkija pystyisi itse seuramaan toimituksen aikana, että on tuottamassa hyväksymiskriteerit täyttävät tulokset. Tässä kohdin on tärkeää korostaa proaktiivisen toiminnan merkitystä, koska reaktiivisen toiminnan myötä alihankintaa teettävä osapuoli voi tuki palauttaa työn tulokset korjattavaksi, mutta menettää samalla kalenteriaikaa muun kokonaisuuden näkökulmasta. Tämän perusteella alihankintaa ostavan, sopivan sekä työnohjauksesta tai muusta seurannasta vastaavan tahon tulisi olla myös tietoinen ohjelmistometriikasta ja siihen liittyvästä sovelmismahdollisuudesta.

2.8 Ohjelmistometriikan jalkautukseen liittyviä muutostekijöitä

Muutostilanteissa esimiehen tulee huomioida yksilöiden ja ryhmien käyttäytymisen. Muutoksen keskellä toiminta ja niiden läpivienti voi onnistua vain, mikäli esimies tuntee ja huomioi riittävän hyvin ihmisten käyttäytymistä ohjaavia periaatteita. Muutoksen aikana esimies johtaa muutosta, mutta on myös samalla itse osa muutoksen kohdetta. Onnistumisen kannalta keskeistä on myös, miten esimies onnistuu kommunikoimaan ymmärrettävällä tavalla muutoksen tarkoituksesta ja tavoitteista, tärkeydestä ja hyödyllisyydestä. (Järvinen 2001, 97-102.)

Muutosten johtaminen vaatii aikaa, rahaa, taitoa, energiaa sekä pitkäjänteisyyttä pysyvien muutosten läpiviemiseksi. Kehittämistehtäviin tulee suhtautua kriittisesti, jotta toimenpiteet helpottaisivat sekä tehostaisivat työntekeä sen sijaan, että kehittämisen seurauksena työntekeäminen hankaloituisi. Tässä yhteydessä puhutaan myös yrityskulttuurin puitteista sekä kehityssuunnasta, mitkä määrittävät yrityksen hallitus sekä ylin johto, joiden kautta muutos säteilee esimiesten kautta koko organisaatioon. (Järvinen 2001, 121-124.)

Ohjelmistometriikan jalkauttamisen haasteena on eri rooleissa toimiville ihmisille riittävän konkreettisen hyödyn osoittaminen niin liiketoiminnallisella, operatiivisella kuin myös henkilökohtaisella tasolla. Ilman todellista ja välitöntä hyötyä on riskinä, että ihmiset eivät koe mielekkääksi ottaa käyttöön uusia asioita.

Toiminnallisten rakenteiden tulee olla selkeitä ja työntekijöillä tulee olla myös kokonaiskuva organisaatiosta, toiminnan organisoimisesta yrityksessä ja työprosessien kulusta, jotta työyhteisö ja sen jäsenet voivat toteuttaa perustehtävänsä sekä saavuttaa sille asetetut tavoitteet. (Järvinen 2001, 32-34.)

Ohjelmistometriikan käyttöönoton ja siihen liittyvän jalkauttamisen osalta tarvitaan seuraavia tekijöitä:

1. johdon sitoutuminen ja tuki
2. budjetti
3. riittävästi työ- ja kalenteriaikaa
4. kehittää toimintamallia niin tehtävien kuin roolien osalta
5. upottaa ohjelmistometriikka osaksi toimintajärjestelmään täydentämällä olemassa olevia kuvauksia sekä ohjeita
6. työvälineiden ja lisenssien hankinta
7. koulutuksien suunnittelu
8. koulutuksien toteutus eri roolien näkökulmasta
9. asioiden toistamista ja kertaamista ajan kanssa
10. menetelmä- ja välinetuki

2.8.1 Asiantuntijatyön luonne

Asiantuntijapalveluissa ratkotaan moniulotteisia ongelmia sekä samalla kehitetään myös uutta. Asiantuntijapalvelut ovat poikkeuksellisen tiedon, taidon sekä tunteiden kauppaamista, minkä ydin on asiakkaan ja neuvonantajan välinen luottamussuhde. Asiantuntijat tuottavat asiakkaalle luvatut palvelut ja he ovat asiantuntijatehtäviin liittyvän koulutuksen saaneita henkilöitä (Sipilä 1998, 12-13.) Tietotyöhön kytkeytyy pääsääntöisesti parempi palkka, mahdollisuus joustoihin, mahdollisuus suunnitella omia töitä sekä tavallista vahvempi työmarkkina-asema. Tietotyön varjopuolena ovat suuret työmäärät, kiire, aikataulupaineet, henkinen rasitus, töiden kasaantuminen sekä työn psyykkinen kuormittavuus. Tietotyöhön liit-

tyy sankarimyynti sekä tietotyön luonnetta kuvaava ristiriitaisuus. (Vesa 2011, 15-16).

Asiantuntija työ edellyttää jatkuvaa oppimista sekä asioiden soveltamista niin yhdessä kuin yksilönä erilaisissa tilanteissa. Asiantuntijan tulee kyetä oppimaan nopeasti laajoja ja monimutkaisia osa-alueita oman arkipäivän työnsä tekemisen yhteydessä. Tästä syystä uuden oppimiselle tulee olla esimiehen tuki. Oppimista voi nopeuttaa riittävän hyvin valmistellulla koulutuksella, minkä tulisi olla kohdennettu yrityksen toimintojen tai roolituksien näkökulmiin soveltuviksi.

Asiantuntijatyön haasteena on tyypillisesti asiakkaan tarpeiden ainutlaatuisuus, missä asiantuntijan tulisi täyttää asetetut odotusarvot sekä tuottaa asiakkaalle hyötyä omalla yritykselle liiketoiminnallisesti kannattavalla tavalla. Tämä johtaa helposti tilanteeseen, missä kaupallisesti tulisi osata ennalta tunnistaa tarvittava työt, arvioida tarvittavan työmäärä sekä lopputulokset ennen kuin kohdealueeseen on ollut mahdollista tutustua. Tämä johtaa helposti ristiriitoihin asiantuntijatyöstä tehtävän tarjouksen, sopimuksen sekä työhön osallistuvien ihmisten välillä.

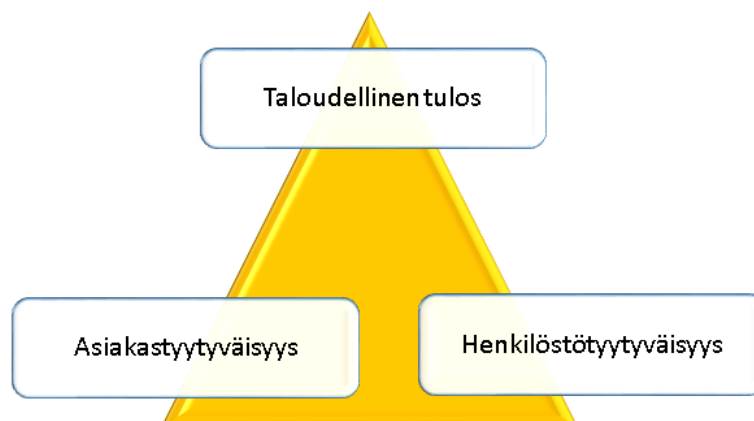
Asiakkaiden vaatimukset palveluiden tuottamiselle kuin myös sisällölle ovat usein ristiriitaisia siihen hintaan nähden, minkä asiakas on valmis maksamaan. Laajat, monimutkaiset järjestelmät sekä niihin liittyvät palvelut pitäisi käytännössä tuottaa taloudellisesta näkökulmasta pienellä henkilömäärällä. Tässä tulee vastaan johtamistapa, yhdessä tekemisen malli sekä työyhteisön keskinäinen yhteentoimivuus. Edellä kuvattujen haasteiden osalta ohjelmistometriikasta saattaa löytyä apuja tehostamaan toimintaa sekä pyrkiä minimoimaan käytännön arkipäivän haasteita, kuten esimerkiksi ”musta tuntuu mittauksia”. Tällöin ohjelmistometriikasta löytynee hyötyä projektille kuin myös henkilötasolla.

3 TIETOJÄRJESTELMÄTOIMITUS KOHDEYRITYKSESSÄ

Tutkimuskohteena oli **kohdeyritys X**, joka on kansainvälisen IT-alan konsernin Suomen maayhtiö. Tässä luvussa esittelen kohdeyrityksen toimintajärjestelmää mukailevat mallit tietojärjestelmän toimittamisesta, projektitoimituksesta sekä tietojärjestelmän rakentamiseen liittyvistä työympäristöistä sekä työvälineistä. Luvun tarkoituksena on luoda konteksti ohjelmistometriikan sijoittumisesta kohdeyrityksen toimintaan.

3.1 Kohdeyrityksen mittaamiskulttuuri

Kohdeyrityksen johtamisen tukena on kolme päätason mittaria: taloudellinen tulos, asiakastytyytyväisyys sekä henkilöstötyytyväisyys. Toiminnan peruseriaatteenä johtamisen näkökulmasta on saavuttaa asetetut tavoitteet siten, että johtamisen kolmiossa (kuva 10) mainitut mittarit ovat keskinäisessä tasapainossa. (Kohdeyrityksen toimintajärjestelmä 2013.)



Kuva 10: Kohdeyrityksen johtamisen kolmio (Kohdeyrityksen toimintajärjestelmä 2013.)

Kohdeyrityksessä mitataan systeemaattisesti toiminnan eri tasoilla johtamisen kolmion mittareihin vaikuttavia tekijöitä. Kohdeyrityksen toiminnassa korostuvat asetettujen tavoitteiden saavuttamisen jatkuva seuranta sekä tarvittaessa korjaavien toimenpiteiden suorittaminen. Tämä koskee myös ohjelmistotoimituksia sekä projekteja. Tyypillisiä tietojärjestelmätoimituksien mittareita ovat liikevaihto, kustannukset, henkilötöyön määrä, kalenteriaika, virheiden sekä muutospyyntöjen lukumäärä. Usein tietojärjestelmän käyttöönoton jälkeen palvelutuotannon aikana

mitataan käyttäjien tukipalveluihin tekemien palvelupyyntöjen lukumäärä, tukipyyntöjen ratkaisuun kulunut keskimääräinen työaika, kustannukset, liikevaihto, ohjelmistovirheiden lukumäärä, käyttökatkojen lukumäärä sekä kesto. (Kohdeyrityksen toimintajärjestelmä 2013.)

Käsitykseni mukaan kohdeyrityksessä ohjelmistometriikka on vähän hyödynnetty apuväline ohjelmistotoimituksien tai ohjelmistoliiketoiminnan johtamisessa. Kohdeyrityksessä ohjelmistometriikkaa hyödynnetään apuvälineenä satunnaisesti osana ohjelmistoratkaisuiden toimittamista lähinnä ohjelmistokehittäjien keskuudessa. Syy tähän löytynee ohjelmistometriikkaan liittyvästä tietämättömyydestä, osaamattomuudesta sekä asenteista. Kohdeyrityksessä ohjelmistometriikka on mielletty ohjelmistokehittäjien työvälineeksi, jolloin liiketoiminnasta vastaavien, päälliköiden, esimiesten tai johdon osalta ei ole riittävästi mielenkiintoa asian tiimoilta.

Oman kokemukseni mukaan kohdeyrityksessä ohjelmistotoimituksien osalta tapahtuva mittaaminen on pääasiassa *reaktiivista*. Mielestäni kohdeyrityksessä tulisi kiinnittää huomiota enemmän *proaktiivisiin* toimintamalleihin, mittaamiseen sekä päätöksiin vaikuttamisen keinoihin käytännön tasolla. Ohjelmistometriikan kytkeminen tiiviimmin ohjelmistoliiketoiminnan johtamiseen osana mahdollistaisi täsmällisemmän ohjelmistojen teknisen laadun varmistamisen sekä parantaisi mahdollisuutta ohjelmistoteknisen laatupoikkeamien havaitsemisesta aikaisessa vaiheessa ohjelmistokehityksen yhteydessä. Kohdeyrityksen näkökulmasta tarve oli rajatulle määrälle keinoja, joiden avulla on mahdollista tehdä tämä tulkinta objektiivisesti henkilöstö riippumatta.

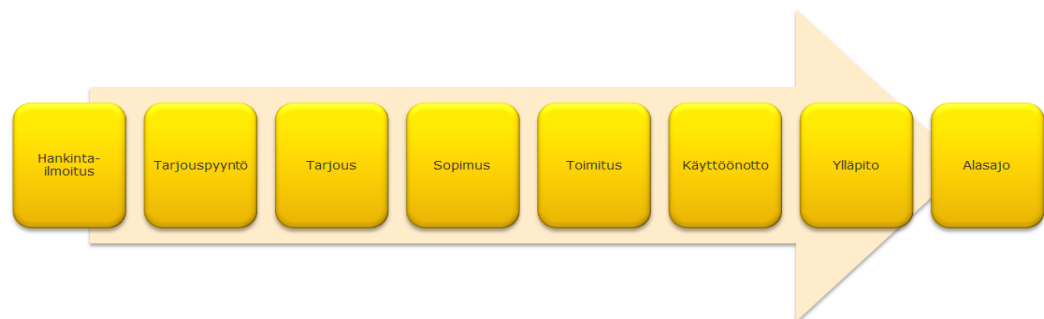
3.2 Kohdeyrityksen tietojärjestelmän toimitusmalli

Kohdeyrityksessä asiakaskohtaisesti räätälöityjen tietojärjestelmien ohjelmistoliiketoiminta on haastava ohjelmistoliiketoiminnan osa-alue. Asiakkaalle mittatilaustyönä tuotettavien ratkaisuiden ja palveluiden liiketoiminnan luonne perustuu asiakaskohtaisten tarpeiden täyttämiseen sekä yhteistyöhön asiakkaan ja mahdollisten kumppanien kanssa. Asiakaskohtaisten ratkaisuiden toimittaminen edellyttää asiakkaan toimialan sekä toiminnan tuntemista, henkilötyöhön painottuvan

erityisosaamisen ja työn hallintaan. Tämän ohjelmistoliiketoiminnan osa-alueen toiminnan erityishaasteena on lukuisten asioiden yhdistäminen sekä soveltaminen asiakkaan liiketoiminnan tarpeiden sekä ohjelmistoteknologian välillä annetuissa aikaulu- ja budjettiraameissa.

Asiakaskohtaisen tietojärjestelmän rakentaminen on kärjistetyksi ”käsityötä”, mitä kohdeyrityksessä on pyritty automatisoimaan tai tukemaan sekä tehostamaan tekemistä soveltuvien työvälineiden avulla. Toiminnan tehostamisen päämääränä on vapauttaa resursseja mekaanisesta työstä, jolloin vapautuva aika olisi kohdennettavissa haasteellisempiin sekä merkityksellisempiin tehtäviin.

Kohdeyrityksessä asiakaskohtaisesti räätälöidyn tietojärjestelmän toimitus on käytännössä liiketoimintaprosessi. Tämän prosessin avulla viedään systeemaattisesti läpi kaupalliset toimenpiteet, rakennetaan sekä toimitetaan asiakkaalle myyty tietojärjestelmä. Tietojärjestelmän käyttöönoton jälkeen tietojärjestelmä siirtyy kehityksestä ylläpitoon sekä jatkuvien palveluiden vastuulle. Liiketoimintaprosessi noudattelee kuvassa 11 esitettyjen prosessien kokonaisuutta (Kohdeyrityksen toimintajärjestelmä 2013.)



Kuva 11: Kohdeyrityksen tietojärjestelmän toimitusmalli (Kohdeyrityksen toimintajärjestelmä 2013.)

1. **Hankintailmoitus, tarjouspyyntö, tarjous ja sopimus** ovat kaupallisia prosesseja, joiden aikana pyritään myymään asiakkaalle tarpeet täyttävän järjestelmän rakentamiseen sekä toimittamiseen liittyvä suuri työkokonaisuus. Toimittajan näkökulmasta kysymys on myös liiketaloudellisesta kannattavuudesta sekä riskien hallinnasta. Nämä vaiheet kestävät tyypillisesti päivistä jopa vuosiin riippuen ratkaisun laajuudesta, toimialasta sekä kohdealueeseen liittyvistä käytännöistä ja kilpailutuksista.

2. **Toimitus** seuraa kaupallisia vaiheita. Toimituksen aikana toimittaja toteuttaa asiakkaalle räätälöidyn tietojärjestelmän tyypillisesti projektina. Toimitus käsittää yleistettynä ratkaisun määrittelyn, suunnittelun, toteutuksen sekä testauksen. Tämän vaiheen aikana räätälöity tietojärjestelmä toteutetaan valmiiksi ohjelmistona, todennetaan toimivuus sekä hyväksytään käyttöönotettavaksi.
3. **Käyttöönotto** seuraa toimitusta ja käsittää pääasiassa asiakkaan henkilöstön kouluttamisen uuden tietojärjestelmän hyödyntämistä varten, tietojen siirtämisen tai konvertoinnin uuteen järjestelmään ja monia muita toimenpiteitä riippuen asiakkaasta, toimituksesta sekä projektista.
4. **Ylläpito** on käyttöönoton jälkeinen vaihe, minkä aikana asiakas hyödyntää ostamaansa tietojärjestelmää oman liiketoimintansa tukena. Ylläpidon osalta toimittajan tehtävänä on tuottaa jatkuvia palveluja asiakalle tietojärjestelmän hyödyntämisen tueksi sekä pitää tietojärjestelmä toimintakuntoisena. Tyypillisiä ylläpitotehtäviä ovat esimerkiksi tietojärjestelmään liittyvien muutoksin, korjauksien, parannuksien sekä päivityksien toimittaminen asiakkaalle.
5. **Alasajo** on tietojärjestelmän käytöstä luopumisesta joko kokonaan tai tuotantokäytössä olleen tietojärjestelmän korvaamisella uudemmalla tietojärjestelmällä.

(Kohdeyrityksen toimintajärjestelmä 2013.)

Asiakaskohtaisten tietojärjestelmien käyttöikä on tyypillisesti vuosia tai vuosikymmeniä. Todellisuudessa tietojärjestelmän elinkaari on tässä tutkimuksessa esitettyä mallia monimutkaisempi. Tietojärjestelmiin kohdentuu vuosien aikana lisämyyntiä erityisesti tietojärjestelmän sovittamisen osalta muuttuneiden tarpeiden, kohdealueen toimintaympäristömuutoksien tai ohjelmistoteknologioiden uudistamistarpeiden takia. Näistä syistä tietojärjestelmän elinkaaren kokonaiskustannukset kasvavat vuosien saatossa suuriksi.

Kohdeyrityksen kannalta mielenkiintoinen näkökulma on, että missä tietojärjestelmän toimitusmallin vaiheissa tulisi hyödyntää ohjelmistometriikkaa ja millä tavalla. Tyypillisesti ohjelmistokehitykseen liittyvät ajatukset, menetelmät sekä työvälineet mielletään kuuluvan ohjelmistoprojektin toteutusvaiheeseen. Tästä

syystä ohjelmistometriikan soveltamista on syytä harkita tutkittavan tietojärjestelmän koko elinkaaren näkökulmasta. Tämä on tarpeen, koska esimerkiksi ohjelmistotoimituksen osalta tarjous, sopimus sekä alustava projektisuunnitelma luovat toimintaedellytyksen toimitukselle. Ohjelmistometriikan osalta tulee tunnistaa tarpeelliset yhtymäkohdat muiden prosessien osalta, jotta ohjelmistometriikan soveltamisen johtamisen apuvälineenä olisi mahdollista upottaa osaksi muuta toimintaa.

3.3 Kohdeyrityksen ohjelmistoprojektin toimitusmalli

Kohdeyrityksessä on selkeästi ohjeistetut projektitoimitusmallit. Lisäksi projektitoimituksen näkökulmasta kohdeyrityksellä on kattavat menetelmät ja käytännöt projektin sekä taloudellisen tilanteen seurantaan varten.

Kohdeyrityksessä on käytössä suuri määrä projektitoimitusmalleja erilaisten ratkaisuiden toimittamiseksi asiakkaalle. Kaikki tietojärjestelmien toimitukset yksittäisenä projektina etenevät kuvassa 12 esitetyn mallin mukaisesti. (Kohdeyrityksen toimintajärjestelmä 2013.)



Kuva 12: Kohdeyrityksen yksinkertaistettu projektimalli. (Kohdeyrityksen toimintajärjestelmä 2013.)

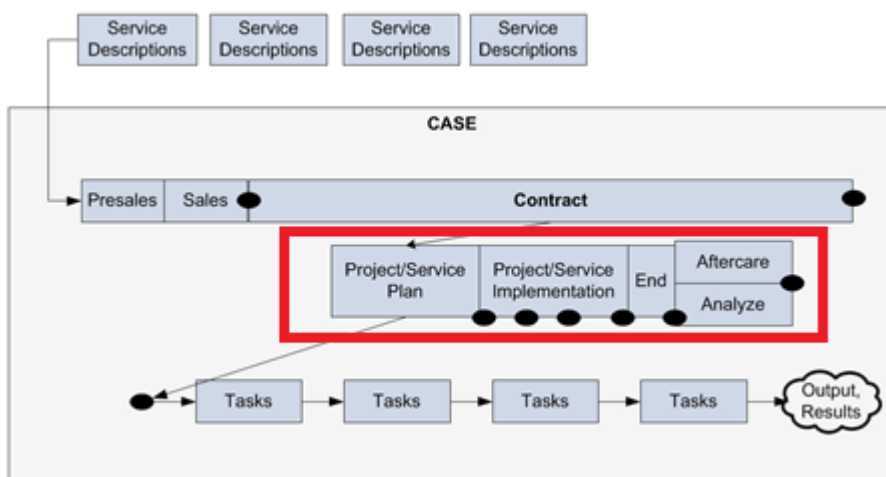
1. **Valmistelun** aikana luodaan projektisuunnitelma sisältäen mm. tehtävät, aikataulu-, työmäärä- ja kustannusarviot sekä suunnitellaan projektiorganisaatio. Tässä yhteydessä tehdään tyypillisesti projektin taloudelliseen kannattavuuteen liittyviä laskemia ja päätöksiä sekä aloitetaan riskien hallinta.
2. **Käynnistyksen** aikana luodaan projektin toimintaedellytykset, työmaa ja toimintaympäristö. Lisäksi käynnistykseen liittyy tyypillisesti projektin käynnistystilaisuus projektin esittelemiseksi tarvittaville osapuolille.
3. **Toteutuksen** aikana toteutetaan projektisuunnitelmaan kirjatut tehtävät haluttujen lopputulosten saavuttamiseksi. Toteutuksen aikana projektia

johdetaan, seurataan sen edistymistä kannalta sekä suoritetaan tarvittavia korjaavia toimenpiteitä poikkeamien osalta.

4. **Päättäminen** on vaihe, minkä aikana projekti lopetetaan, puretaan projektiorganisaatio sekä projektin työmaa. Projektin päättämiseksi projektista tuotetaan loppuraportti.
5. **Jälkitoimet** edustavat tyypillisesti varsinaisesta projektista riippumattomia kohdeyrityksen sisäisiä toimenpiteitä, kuten esimerkiksi analysointi tapah-
tuneesta, opitun ja kokemustiedon kerääminen muita käyttötarkoituksia varten.

(Kohdeyrityksen toimintajärjestelmä 2013.)

Kohdeyrityksessä projekti rakentuu projektitoimitusprosessimallin mukaisesti, mikä määrittelee yhdessä laatujärjestelmän kanssa johtamisen kannalta tarvittavat päätökset kuvassa 13 esitetyt mustat pisteet ennen kuin mahdollista siirtyä seuraaviin työvaiheisiin. Kohdeyrityksen toimintajärjestelmä määrittelee myös katselmoinnit sekä muut laadulliset toimenpiteet, joiden avulla varmistetaan projektitoimituksen laatu. (Kohdeyrityksen toimintajärjestelmä 2013.) Ohjelmistometriikan soveltaminen tulisi kytkeä osaksi projektin ohjauskäytäntöjä. Lisäksi projektin näkökulmasta ohjelmistometriikka tulisi liittää osaksi katselmointeja sekä ennen kaikkia työmenetelmiä ja työskentely-ympäristöjä.



Kuva 13: Kohdeyrityksen malli sopimuksen ja projektitoimituksen yhteydestä (Kohdeyrityksen toimintajärjestelmä 2013.)

Ohjelmistoprojektin suhde myyntiin, tarjoukseen ja sopimukseen on esitetty kuvassa 13, missä projekti on korostettu punaisella paksulla suorakulmiolla. Myynti

hyödyntää palvelukuvauksia, mitkä kuvaavat asiakkaalle tuotettavat palvelut, hyödyt sekä toimintamallin palvelun tuottamiseksi. Myynnin aikana asiakkaalle jätetään tarjous toimitettavasta tietojärjestelmästä sisältäen alustavan ratkaisun kuvauksen sekä luonnoksen projektisuunnitelmasta. Sopimusneuvottelun jälkeen toimitettavasta tietojärjestelmästä allekirjoitetaan sopimus asiakkaan kanssa. Varsinainen tietojärjestelmän tuottava projekti toteutetaan projektisuunnitelman mukaisesti.

Projektin valmistelu tapahtuu osittain asiakkaalle tarjottavan ratkaisun tarjousvalmistelun ja sopimusneuvottelun aikana. Projektin valmistelun aikana syntyy joukko tunnuslukuja, joiden perusteella tehdään päätös ratkaisun tarjoamisesta asiakkaalla. Näiden tunnuslukujen perusteella seurataan myös projektin etenemistä sekä tarvittaessa suoritetaan korvaavia toimenpiteitä, luodaan ennusteita jäljellä olevasta työstä sekä ennen kaikkea tehdään päätöksiä jatkotoimenpiteistä. (Kohdeyrityksen toimintajärjestelmä 2013.) Edellä kuvatun pohjalta olisi luonnollista kytkeä ohjelmistometriikan soveltaminen projektin tavoitteisiin sekä tilanneraportoinnin kautta osaksi projektin johtamista. Käytännössä toimiakseen tämä vaatisi, että mittauskäytäntö on selkeä, säännöllinen sekä myös vastuutettu projektissa tehtäväksi soveltuvilla työvälineillä.

Räätälöidyn asiakaskohtaisen tietojärjestelmän toimitusprojektin valmistelun, toimituksen sekä päättämisen yhteydessä on luontevaa arvioida kokonaisuutena toimitettavan tietojärjestelmän ohjelmistoratkaisun laajuutta sekä ylläpidettävyyttä lukumääriin perustuvien olettamuksien ja tunnuslukujen avulla. Projektitoimitusmallin määrittelemisissä päätöksentekopisteissä tai laadun varmistamiseen liittyvien katselmointien yhteydessä on myös selkeät kohdat laajentaa tarvittavilta osin kohdeyrityksen prosesseja, kuvauksia sekä ohjeita.

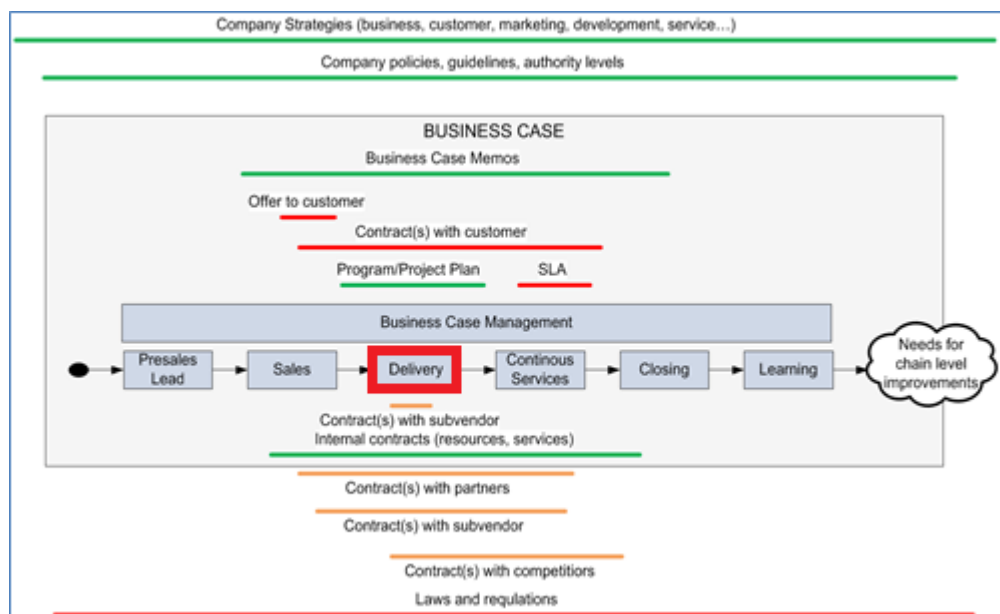
3.4 Kohdeyrityksen tietojärjestelmätoimituksen Business Case

Asiakaskohtaisesti räätälöityjen tietojärjestelmien sekä niihin liittyvien palveluiden liiketoiminnan luonteelle on tyypillistä harvoin tapahtuvat kaupat, koska kyseessä on tyypillisesti asiakkaan näkökulmasta varsin mittavasta inversoinnista, mihin liittyy riskejä. Lisäksi lukuisten virallisten ja de-facto standardien sekä par-

haiden käytäntöjen osalta prosessien, menetelmien ja työtapojen määrä on hyvin suuri. Asiakaskohtaisesti räätälöitävien tietojärjestelmien myynnin, kehittämisen, toimittamisen sekä palvelutuotannon osalta asiakkaiden vaikutus käytäntöihin on merkittävä.

Käytännössä ohjelmistoliiketoiminnan näkökulmasta räätälöidyn tietojärjestelmän toimittaminen projektina on vain yksi osa laajempaa kokonaisuutta. Laajojen räätälöityjen asiakaskohtaisten tietojärjestelmien toimitusprojektin ympärille liittyy tyypillisesti useita sidosryhmiä, myyntiprojekteja, sopimuksia sekä prosesseja. Tämän tutkimuksen tarkastelukohteena oleva räätälöidyn tietojärjestelmän toimitaminen projektina liittyy kokonaisuuteen, mitä voidaan kutsua termillä **business case**. Kohdeyrityksen näkökulmasta business casen tulee olla taloudellisesti riittävän kannattava siten, että myös asiakas- sekä henkilöstötyytyväisyys säilyvät yrityksen johdon odotuksien mukaisella tasolla.

Kuvassa 14 olen kuvannut yksinkertaistettuna yksittäisen business casen ja siihen tyypillisesti liittyviä sidosryhmiä, sopimuksia, prosesseja sekä toimitusprojektin projektinsuunnitelman sijoittumista osaksi kokonaisuutta. Kuvassa prosessit alkavat vasemmalta ja etenevät ajallisesti oikealle.

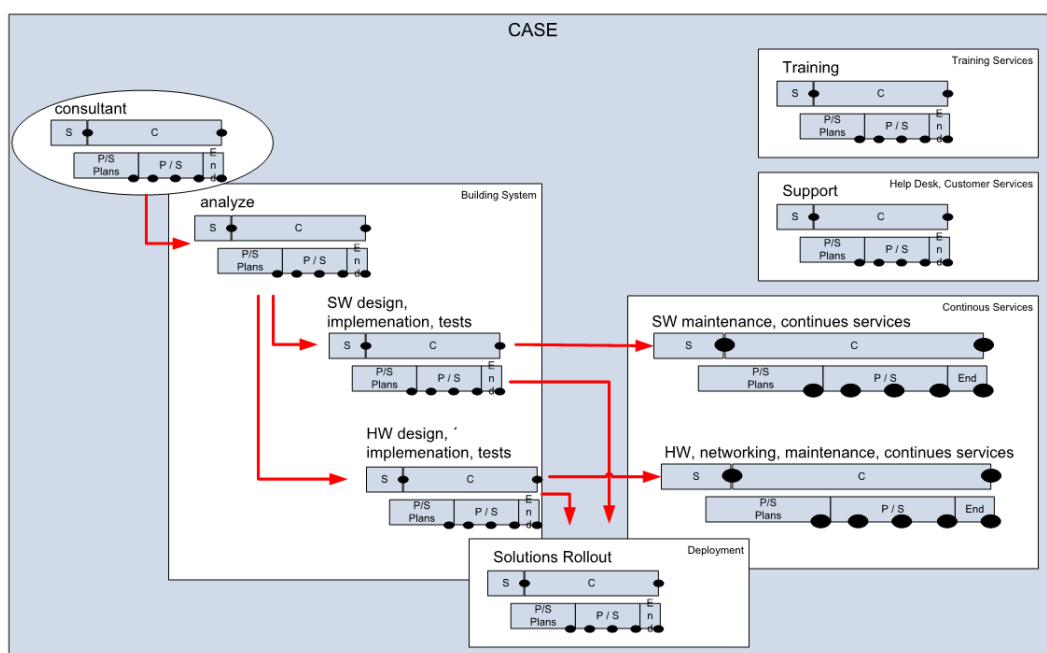


Kuva 14: Kohdeyrityksen Business Case -malli (Kohdeyrityksen toimintajärjestelmä 2013.)

Johtamisen sekä mittareiden hyödyntämisen näkökulmasta business case luo osaltaa erinomaisen kehyksen tarkastella ohjelmistometriikan hyödyntämistä sekä

toiminnallisella sekä yksilötasolla. Erityisesti business casen näkökulmasta ohjelmistometriikan mahdollisia soveltamiskohteita ovat tietojärjestelmän toimitusprojektin lisäksi myynti, sopimus, palvelulupaukset, kumppanit, alihankkijat, projektisuunnitelma, jatkuvat palvelut sekä myös oppiminen jo tehdystä työstä. Yksilötasolla ohjelmistometriikan tarkastelu on mahdollista tehdä prosesseihin liittyvien roolien kautta.

Laajan asiakaskohtaisen räätälöidyn tietojärjestelmän toimittaminen asiakkaalle etenee usein kuvassa 15 yksinkertaistetun mallin mukaisesti useiden projektien sekä sopimuksien muodossa. Tyypillisesti toimitus muodostuu yleensä yhden tai useamman toimittajan eri projektista sekä samalla myös asiakkaan sisäisistä toimintamallin kehitys- sekä uuden toimintamallin jalkautusprojekteista. Laajempaa kokonaisuutta voisi asiakkaan näkökulmasta luonnehtia asiakkaan kehityshankkeeksi, minkä kokonaiskesto on tyypillisesti vuosia.



Kuva 15: Malli asiakaskohtaisen tietojärjestelmän toimituksesta. (Kohdeyrityksen toimintajärjestelmä 2013.)

Kokonaisuuden näkökulmasta räätälöidyn tietojärjestelmän toimitus voi jakaantua vaiheisiin tai osaprojekteihin. Kuva 15 esittää kohdeyrityksen näkökulmasta tyypillisiä asiakaskohtaisen tietojärjestelmän toimitukseen sisältyviä osaprojekteja: konsultointi, määrittely, ohjelmiston suunnittelu ja toteutus, käyttöympäristöjen perustaminen sekä käyttöönotto. Näiden lisäksi keskeisiä muita kokonaisuuksia

ovat koulutus, tuki ja ohjelmistojen ylläpito sekä käyttöympäristöjen ylläpito. Kaikkiin osakokonaisuuksiin liittyvät tarjous, myynti, sopimus, suunnittelu, aloitus, toteutus, päättäminen, katselmoinnit sekä myös johtaminen, seuranta sekä mittaaminen. (Kohdeyrityksen toimintajärjestelmä 2013.)

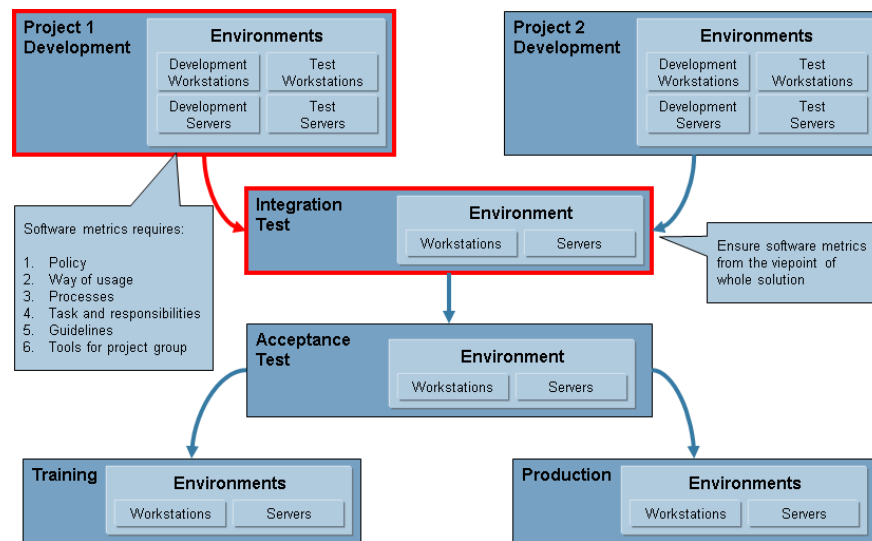
Laajasta mittakaavasta tarkasteluna kuvassa 17 ohjelmistometriikka sijoittuu ohjelmiston suunnitteluun ja toteutukseen sekä myös ohjelmiston ylläpitoon ja jatkuviin palveluihin. Ohjelmiston suunnittelu- ja toteutusprojektin tekninen onnistuminen on liiketoiminnallisesti kriittistä, koska viivästymiset ja ohjelmiston heikko laatu vaikuttavat välittömästi negatiivisesti jatkoprojekteihin sekä aikanaan ylläpitoon sekä jatkuviin palveluihin. Tässä yhteydessä on tärkeää korostaa, että aikaisempien vaiheiden tai projektien ongelmien kerrannaisvaikutukset ovat suuret edellä kuvatun kaltaisessa kokonaisuudessa. Yksinkertaistettuna ohjelmiston heikon teknisen laadun pieni parantaminen saattaa vastaavasti johtaa merkittäviin kustannussäätöihin. Tältä pohjalta väitän, että ohjelmistometriikan roolia tietojärjestelmän toimituksen johtamisen apuvälineenä ei tule väheksyä.

3.5 Kohdeyrityksen palveluiden tuottamisympäristöt

Kohdeyrityksessä käytössä olevat useat ohjelmistokehitykseen tarkoitetut työvälineet tukevat käytännön tasolla vaihtelevasti ohjelmistometriikkaa, kuten esimerkiksi Microsoft Visual Studio. Käytännössä omien havaintojeni mukaan kohdeyrityksen käytännön arkipäivän työssä ei systemaattisesti hyödynnetä ohjelmistometriikan työvälineitä. Tämä johtuu syystä, että tietojärjestelmätoimituksien tai projektien johtamisen osalta ei systemaattisesti edellytetä ohjelmistometriikan soveltamista.

Ohjelmistometriikkaan liittyvä hyödyntäminen, toimintatavat, menetelmät sekä työvälineet liittyvät keskeisesti ohjelmiston tekniseen toteutukseen ja testaukseen. Tästä syystä ohjelmistometriikan osalta sen hyödyntämiseen liittyvät linjaukset, ohjeistukset, tehtävät sekä työvälineet liittyvät olennaisesti ohjelmistokehitysympäristöön sekä integrointitestiympäristöihin, mitkä ovat korostettu punaisella kuvassa 16. Ohjelmistojen kehitystyö (ohjelmointi sekä lähdekoodin kirjoittaminen) työ tapahtuu kehitysympäristössä (development environment). Ohjelmiston valmistuneet osat, komponentit, modulit tai kirjastot integroidaan osaksi muuta

kokonaisuutta integrointitestiympäristössä ennen ohjelmiston toiminnallista testaamista.

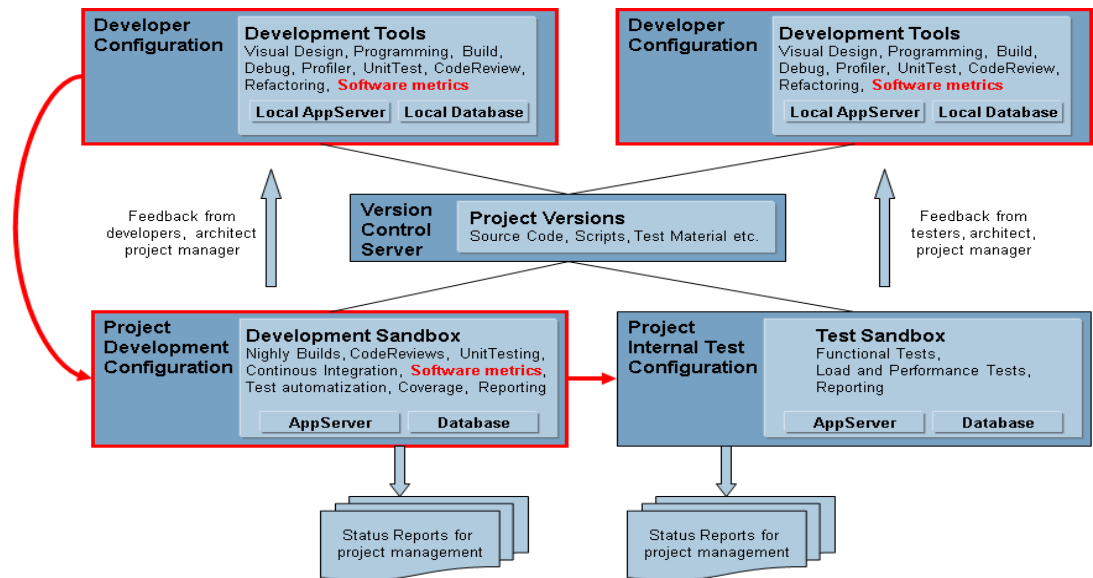


Kuva 16: Ohjelmistometriikka sijoitettuna kohdeyrityksen työympäristöön

Käytännössä ohjelmistokehitysympäristössä tulee tapahtua tarvittavat toimenpiteet ennen ohjelmistokehitystä sekä ohjelmistokehityksen aikana tai muuten ohjelmisto-osuuden edellyttämällä tavalla. Tästä syystä korostuu tarve työn johtamisen puolelta asettaa mitattavat tavoitteet ennen työn tekemistä. Työn tekemisen aikana tulee myös valvoa, että sovittuja käytäntöjä noudatetaan sekä lisäksi tarvittavat korjaavat toimenpiteet suoritetaan välittömästi poikkeaman tullessa esiin. Työn tekemisen jälkeen tulee myös todentaa, että syntyneet lopputulokset vastaavat asetettuja tavoitteita. Lopputulosten todentaminen voi tapahtua testaamisen yhteydessä, jolloin useamman kehittäjän tai useamman projektin tuottamat lopputulokset voidaan mitata sovittujen käytäntöjen mukaisesti kerralla esimerkiksi ohjelmiston integraatiotestiympäristössä.

Työvälineiden osalta työnohjauksen tai johdon tulee huolehtia siitä, että henkilöllä on käytettävissä tarvittavat ohjeistus, käsitys toimintamallista, tavoitteista, seurannasta, todentamista sekä käytännön vastuista projekti- tai ohjelmistoratkaisu-kohtaisesti. Lisäksi ohjelmistokehittäjät sekä testaajat tai muut laatuvaastavat tarvitsevat käyttämiinsä ympäristöihin soveltuvat työvälineet. Johtamisen vastuulla on vaatia näitä asioita sekä tarvittaessa järjestää aikaa, rahaa sekä koulutusta. Kuvassa 17 olen kuvannut ohjelmistometriikan työvälineiden sijoittumista osaksi tietojärjestelmään liittyvää kohdeyrityksen kehitysympäristöä. Projektipäällikön,

laatuvastaavan, arkkitehdin tai esimiehen tulisi mielestäni olla tietoinen työskentely-ympäristöihin liittyvistä tehtäväkokonaisuuksista ja niitä tukevista työvälineistä. Ilman riittävää ymmärrystä työympäristöihin kohdistuvista tarpeista saattaa olla mahdollista, ettei projektin työympäristöihin järjestetä soveltuvia työvälineitä, mitkä mahdollistaisivat ohjelmistometriikan hyödyntämisen.



Kuva 17: Ohjelmistometriikka kohdeyrityksen sovelluskehitysympäristössä

Ohjelmistometriikan soveltaminen edellyttää johtamiselta kuria sekä pysymistä linjassa, vaikka olisi houkutus ohittaa mahdolliset haasteet tai turhalta vaikuttavat työvaiheet. Lisäksi johtamisen osalta tulee kiinnittää huomiota myös siihen, että projektiryhmän puolelta esitettyjä tarpeita tai vaatimuksia korjata asiat kuntoon ei saa lakaista maton alle. Työnohjauksen tai muun johdon tulee tarvittaessa järjestää aika ja paikka korjaaville toimenpiteille mahdollisimman nopeasti ennen siirtymistä eteenpäin projektissa, jotta olisi mahdollista välttyä suuremmilta ongelmilta. Asiana ohjelmistometriikan soveltaminen osana arkipäivän työtä liittyy keskeisesti johtamiseen, koska johtamisen kautta on mahdollista tiedostamatta estää ongelmien korjaaminen tai siirtää asian käsittely tulevaisuuteen.

3.6 Ohjelmistometriikan hyödyntäminen kohdeyrityksen tietojärjestelmätoimituksessa

Tietojärjestelmän kehittämistä edeltävässä kaupallisessa vaiheessa myynnin, tarjouspyynnön käsittelyn, tarjousvalmistelun sekä sopimusneuvottelun aikana oh-

jelmistometriikkaa olisi mahdollista hyödyntää eri tavoin. Normaalisti kohdeyrityksessä tarjoukseen sekä sopimukseen kirjataan tavoitteita numeerisina arvioina aikataulun, työmäärien sekä kustannusten osalta, joiden avulla toimitusta seurataan sekä johdetaan. Näihin olettamuksiin liittyy tyypillisesti myös pelivara, joiden puitteista voidaan tehdä pieniä muutoksia tai nopeita päätöksiä tilanteen korjaamiseksi tai sopeuttamiseksi.

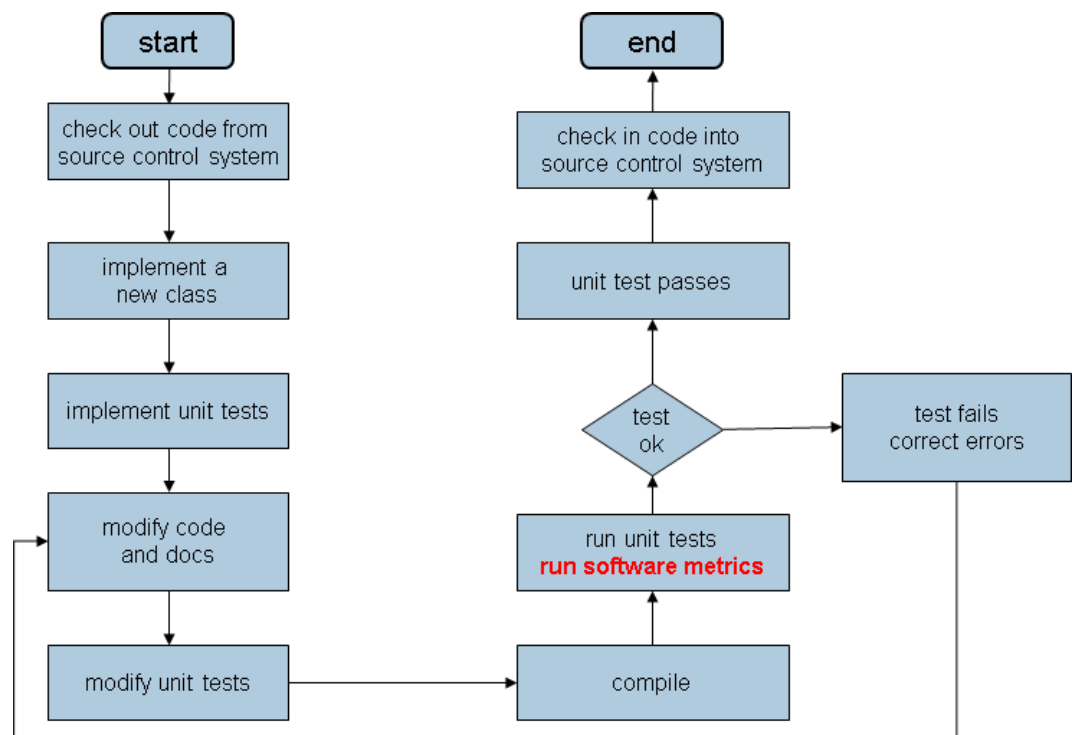
Arvioiden osalta on mahdollista hyödyntää kolmenlaista tietoa: toimialan arviot muista vastaavista mittaushaasteista, organisaation historia tiedot sekä projektikohtainen näkemys. Estimaattien tekemisen haastena on arvion tekijän subjektiivinen näkemys sekä tiedostamaton optimismi. Tähän vaikuttavat merkittävästi projektipäällikön tai muun arvioijan tausta, kokemus sekä vankat uskomukset uudesta paremmasta teknologiasta tai olettamuksesta, että edellisen projektin haasteet eivät toteudu tällä kertaa. Yksittäisen projektin osalta arvioiden lähtökohdaksi on syytä soveltaa oman yrityksen mittaushistoriaa projektin tuottavuusolettamuksien lähtökohdaksi. (McConnell 2006, 91-95.) Ohjelmistometriikan mittaushistorian hyödyntäminen uusien alkavien ohjelmistotoimituksien arvioiden lähtötietona edellyttää soveltuvan mittaustiedon keräämistä projektien aikana sekä projektin päättymisen jälkeen.

Kohdeyrityksessä tulisi kerätä systemaattisesti soveltuvat ohjelmistometriikan mittaustiedot ainakin projektitasolla, jotta ohjelmistometriikasta olisi saavutettavissa hyötyä. Tämä tarkoittaa käytännössä soveltuvien sekä yhteismitallisten mitareiden määrittämistä, mittaustavan, tehtävien sekä vastuiden suunnittelua, mittaustietovaraston perustamista, prosessi- ja toimintaohjeiden täydentämistä tarvittavilta osin. Lisäksi erityisen suuri työ on soveltuva toimintamallin jalkauttaminen organisaatiolle.

Asetettujen ohjelmistometriikan kriteerien avulla olisi merkittävien poikkeamien osalta mahdollista avata toimittajan puolelta keskustelu asiakkaan kanssa kustannusten, aikataulun tai budjetin sovittamisesta uuteen tilanteeseen. Samalla tarjouksen sekä sopimuksen avulla ohjelmistotoimituksesta vastaavaan olisi helpompia sopia sekä seurata toimituksen aikana esimerkiksi alihankkijoiden tai offshore-toimituksien teknistä laatua. Käytännössä asian sivuuttaminen kaupallisessa vaiheessa johtaa helposti tilanteeseen, missä ohjelmistometriikkaa ei sovelleta oh-

jelmistoprojektin aikana tai ainakin ohjelmistometriikan hyödyntäminen jää irralliseksi päätöksen tekemisestä eikä siten päädy osaksi johtamista.

Ennen sovellusten kehittämistä tulee valita soveltuvat mittarit sekä näille mittareille kyseiseen projektiin soveltuvat raja-arvot. Tämä on tärkeää tehdä ennen järjestelmän ohjelmistoteknistä toteutusta, jotta toiminnalle asetettavat raamit tulevat jalkautetuksi koko projektiryhmälle ennen töiden aloittamista. Toinen erityisesti korostettava seikka on ohjeistaa projektiryhmä korjaamaan välittömästi mittareiden osoittamat poikkeamat ennen muiden töiden jatkamista. Etukäteen asetettut mittarit auttavat myös toteuttajia suunnittelemaan toteutustyönä paremmin sekä samalla asettaa myös mahdollisuuden seurata oman työnsä tuloksia välittömästi työntekeksen aikana. Tämä puolestaan mahdollistaa poikkeamien korjaamisen välittömästi niiden syntymisen yhteydessä, jolloin mahdollistetaan kustannus- ja aikataulusäästöt laadullisten poikkeamien korjaamisen osalta jälkikäteen.



Kuva 168: Ohjelmistometriikka osana ohjelmistokehittäjän työtä

Konkreettisesti ohjelmistometriikan soveltaminen tulee upottaa osaksi muuta tekemistä sekä prosesseja. Esimerkiksi sovelluskehittäjän työn näkökulmasta uuden ohjelmisto-osuuden toteutuksen kannalta työprosessi voisi edetä kuvan 18 mukaisesti, jolloin ohjelmistometriikan ajaminen siihen tarkoitetuilla automaattisilla

työvälineillä olisi osa ohjelmistokehityksen työvaiheita. Tässä lähestymistavassa ohjelmistokehittäjän tulee välittömästi korjata toteuttamansa ohjelmistotekninen toteutus vaadittujen ohjelmistometriikan raja-arvojen mukaiseksi, jotta esimerkiksi ohjelmiston rakenteellinen monimutkaisuus ei päädy aiheuttamaan myöhemmissä vaiheissa tarpeettomia laatukustannuksia.

Ohjelmiston ylläpitäjän näkökulmasta erityisenä haasteena ovat muiden tekemät sadat tuhannet tai miljoonat koodirivit, mitkä ylläpitäjän tulisi pystyä tarvittaessa omaksumaan nopeasti sekä kyetä korjaamaan aiheuttamatta häiriöitä tai uusia virheitä. Ohjelmiston toimitusprojektin aikainen kiire tai varsinaisen ratkaisun toteutus vajaavaisilla resursseilla on saattanut johtaa heikkoon laatuun ohjelmistokehityksen aikana. Nämä ongelmat siirtyvät sovelluksen ylläpitäjälle. Ohjelmiston ylläpidon näkökulmasta yksi tyypillinen ongelma on lähdekoodin monimutkaisuus sekä ohjelmiston epäselvä tekninen rakenne. Tällöin kehityksen heikolla laadulla on suoria vaikutuksia ylläpidon kustannuksiin. Tästä syntyy helposti syvenevä ongelmakierre, jolloin ohjelmiston monimutkaisuus kasvaa sekä laatu heikenee entisestään. (Alain 2008, 4-8.)

Tietojärjestelmän toimituksen aikana rakennettavien ohjelmistoteknisen toteutusosuuksien valmistumisen yhteydessä tulisi aina varmistaa ennen ohjelmiston testausta, että tekninen toteutus on noudattanut määriteltyjä linjauksia. Tarvittavat korjaukset tulisi tehdä tai vähintään kirjata nämä poikkeamat perusteluineen sekä jatkotoimenpidesuosituksiin ennen ratkaisun siirtoa sovellusylläpitoon sekä järjestelmän osalta tarjottaviin jatkuviin palveluihin.

Sovelluksen ylläpitäjällä on rajoitetusti mahdollisuuksia vaikuttaa ohjelmistokehitykseen tai siellä syntyvään laatuun. Tästä syystä tulevan ohjelmiston ylläpitäjän tulisi mahdollisimman aikaisessa vaiheessa päästä vaikuttamaan ohjelmiston ylläpidettävyyteen. Yksi vaikuttamiskeino on, että ohjelmistolle asetetaan sen kehittämisen aikana mitattavia tavoitteita, joiden toteutumista seurataan sekä ohjelmiston kehittämisen aikana ja sen jälkeen. Ylläpitoprosessin näkökulmasta esimerkiksi ohjelmiston sisältämä lähdekoodin rivimäärä (LOC) on yksi mahdollinen mitattava sekä seurattava asia. Lisäksi ohjelmiston sisäisen rakenteen monimutkaisuuden mittaaminen on asia, mihin ylläpidosta vastaavien tulisi kiinnittää erityisiä huomioita. Ohjelmiston sisäisen monimutkaisuuden mittaamisella voidaan

arvioida esimerkiksi ohjelmiston ylläpidettävyyttä sekä hyödyntää mittaustietoa ohjelmiston siirtämisestä kehityksen osalta sovellusylläpitoon hyväksymiskriteerinä. Ohjelmiston mittaamiseen liittyviä kaupallisia välineitä on hyvin saatavilla. (Alain 2008, 24-29.)

4 TUTKIMUSKONTEKSTI JA -MENETELMÄT

Tässä luvussa esittelen tutkimukseen sisältyvän tutkimuskontekstin, kyselytutkimuksen periaatteet ja tavoitteet. Luvussa esittelen myös aineiston keruu- ja käsittelymenetelmän.

4.1 Tutkimuskonteksti

Kohdeyritys kuului kansainväliseen suuren IT-alan konserniin, jonka palveluksessa oli tuhansia työntekijöitä. Kohdeyrityksen palvelutarjontaan sisältyi muun muassa konsultointia, ohjelmistoratkaisuiden tuottamista sekä ylläpitoa, ohjelmistoihin liittyviä jatkuvia palveluita, ohjelmistojen integraatioita sekä myös ulkoistuspalveluita. Tutkimuskohteena oli kohdeyrityksen Suomen maayhtiön yksi ohjelmistoliiketoimintayksikkö, minkä palveluksessa on satoja työntekijöitä. Tarkastelin tutkimuskohdetta kohdeyrityksen sisäisestä näkökulmasta ja rajasin tarkastelun yhteen projektina toimitettavan asiakaskohtaisesti räätälöidyn tietojärjestelmätoimitukseen.

Työskentelin tutkimuksen tekemisen aikana kohdeyrityksessä esimiehenä. Tämä mahdollisti minulle tutkimuksen tekemisen osallistumalla kohdeyrityksen arkipäivän toimintaan aidossa ympäristössä. Tutkimuksen tekeminen yrityksen ulkopuolisena tutkijana ilman kohdeyrityksen tuntemusta olisi ollut haasteellinen tehtävä.

4.2 Tutkimusaineiston hankinta

Tutkimukseen sisältyneen sähköisen kyselytutkimuksen tarkoituksena oli tuottaa käsitys ohjelmistometriikan nykytilasta kohdeyrityksessä. Kyselytutkimuksen sisällytettävien tietojärjestelmän toimituksen elinkaaren vaiheiden sekä roolien näkökulmien yhtymäkohtien avulla tarkoituksena oli selvittää mahdollisen erikseen tehtävän kehityshankkeen painopistealueita sekä rajoituksia. Kyselytutkimuksen tehtävänä oli pyrkiä olemaan riittävän avoin sekä löytämään roolien ja tehtävien läpileikkauksien kautta mahdollisia sokeita pisteitä tai yllättäviä näkökulmia.

Laadin kyselytutkimuksen teemat sekä niihin liittyvät kysymykset luomani kokonaiskuvan (kuva 19) kohdeyrityksen tietojärjestelmätoimitusmallin sekä roolien pohjalta. Kyselytutkimuksen teemoiksi valitsin seuraavat tutkimuksen osa-alueet:

1) ohjelmistometriikan osaaminen ja koulutustarve, 2) ohjelmistometriikan työvälineet, 3) ohjelmistometriikan hyödyntäminen sekä 5) ohjelmistometriikan käyttöönotto. Laadin kysymykset edellä mainittujen teemojen, tutkimuksen tavoitteiden sekä tutkimuskysymyksen pohjalta siten, että tuotin yhden kysymyspatterin sisältäen 29 kysymystä.

Kyselytutkimuksen sisällöllisestä tarpeesta johtuen laadin kyselytutkimuksen vastausvaihtoehdot siten, että vastaajat voivat vastaukset numeerisena automaattisen käsittelyn mahdollistamiseksi. Kyselytutkimuksen vastauksien analysoinnin kautta muodostettiin kyselytutkimuksen lopputulokset kokonaisuuden osalta, joiden pohjalta tuotettiin erikseen induktiivisen päättelyn keinoin havainnot sekä johtopäätökset.

Kyselytutkimuksen kysymyksiin laadin vastausvaihtoehdot siten, että kaikkiin kysymyksiin vastaus oli annettava numeerisella arvolla (0-6). Kaikkien kysymyksen osalta asteilla nolla tarkoitti, ettei kysymys koske vastaajaa. Asteikon arvot 1-2 tarkoittivat vähäistä painoarvoa tai heikkoa valmiutta eli käytännössä mahdollisia kehityskohteita. Asteikon arvot 3-4 merkitsivät neutaalia tulosta tai asioita, joita olisi mahdollisesti syytä parantaa. Asteikolla arvot 5-6 tarkoittivat positiivista tulosta tai hyvällä mallilla olevia asioita, joiden osalta kehittämistarve on vähäinen.

Kyselytutkimusta varten toteutin sähköisen kyselylomakkeen (LIITE 3) kohdeyrityksessä käytössä olevan Microsoft SharePoint -ratkaisuun. Kyselytutkimuksen alussa yksikön johtaja toimi koekäyttäjänä, jonka kommenttien perusteella kyselytutkimusta jatkui ilman sisällöllisiä muutoksia.

Kyselytutkimukseen vastaamista varten kohdeyrityksen järjestelmien tietojen perusteella poimittiin tutkimuskohteena olevan yksikön ohjelmistoratkaisuiden parissa toimivat henkilöt. Tavoitteena oli saada kymmenen vastausta jokaista roolia kohden riittävän vastauskattavuuden saavuttamiseksi, mikä tavoitteena tarkoitti 100 vastaajan joukkoa. Riittävän vastausprosessin saavuttamiseksi kysely lähetettiin 130 henkilölle. Kyselytutkimuksen vastaukset kerättiin anonymisti johtuen kohdeyrityksen turvallisuuden hallintajärjestelmän edellyttämästä toimintamallista.

Suoritin kohdeyrityksen järjestelmiä käyttäen poiminnan henkilöistä. Toimitin 15.6.2013 ja 26.8.2013 kutsut (LIITE 1) sähköpostitse osallistua kyselytutkimukseen yhteensä 130 henkilölle, joilta vastauksia palautui 39 kappaletta. Kyselytutkimuksen vastausprosentti oli 30 %, joka oli odotettua matalampi. Alhaista vastausprosenttia selittävät kohdeyrityksen organisaatiossa samalla ajanjaksolle kohdistuneet useat yrityksen sisäiset organisaatiomuutokset, uusien toimintamallien sekä uuden toiminnanohjausjärjestelmän käyttöönotto. Käytännössä kyselytutkimus jäi tärkeämpien asioiden jalkoihin.

4.3 Tutkimusaineiston käsittely ja analysointi

Kyselytutkimuksen sisällön avulla pyrittiin selvittämään vastaajien roolien näkökulmista asiakaskohtaisesti räätälöitävän tietojärjestelmän toimituksen vaiheista. Rooleja olivat johtaja, esimies, myyjä, projektipäällikkö, arkkitehti, määrittelijä, suunnittelija, toteuttaja sekä testaaja. Kyselytutkimuksen avulla tarkasteltiin seuraavia elinkaaren seuraavista vaiheista: hankintailmoitus, tarjouspyyntö, tarjous, sopimus, toimitus, käyttöönotto, ylläpito, alasajo. Tässä yhteydessä vaiheella tarkoitettiin yksinkertaistettua loogista jakoa sekä yleistystä eri prosessien tehtävistä ja näiden yhtymäkohdista liiallisen monimutkaisuuden välttämiseksi.

Kyselytutkimuksen analysoinnin päämääränä olisi tarkastella kyselytuloksia kokonaisuutena roolista sekä tietojärjestelmän elinkaaren vaiheista riippumatta. Analysoinnin osalta linjattiin, että vastauksien analysointi roolien näkökulmasta tehdään, mikäli vastauksia on riittävästi kaikkien roolien osalta.

Siirsin sähköisen kyselytutkimuksen vastaukset SharePointista kohdeyrityksen SQL Server tietokantaan lokakuussa 2013. Käsittelin SQL Serverin avulla kyselytutkimuksen vastaukset sekä tuotin välituloksina kyselytutkimuksen vastaukset (LIITE 4) sekä kyselytutkimuksen vastauksien yhteenveto (LIITE 5). Kyselytutkimuksen vastauksien pohjalta tuotin graafit sekä koosteet (LIITE 6) jokaisen kysymyksen osalta, joiden pohjalta tein kyselytutkimuksen tulosten analysoinnin sekä johtopäätökset.

5 TUTKIMUKSEN TULOKSET

Tässä luvussa esittelen sähköisen kyselytutkimuksen tulokset teemoittain. Lisäksi esittelen kyselytutkimuksesta tekemäni havainnot sekä johtopäätökset.

5.1 Kyselytutkimuksen tulokset

Kyselytutkimukseen vastanneiden (39 kpl) roolit jakautuivat seuraavasti:

- hallinto 13 % (5 kpl): johtaja, esimies, myynti
- työnohjaus 18 % (7 kpl): projektipäällikkö, palvelupäällikkö
- työryhmä 69 % (27 kpl): arkkitehti, määrittelijä, suunnittelija, toteuttaja, testaaja

Kyselytutkimukseen osalta erityinen huomio oli, ettei kohdeyrityksen myyjiltä palautunut vastauksia. Myyjien puuttuminen vastauksista ei ole kriittistä, koska myös johtajat, esimiehet ja projektipäälliköt osallistuvat myyntityöhön.

Sähköisen kyselytutkimuksen kysymyksien teemat olivat ohjelmistometriikan osaaminen ja koulutustarve, ohjelmistometriikan työvälineet, ohjelmistometriikan hyödyntäminen sekä ohjelmistometriikan käyttöönotto.

5.1.1 Ohjelmistometriikan osaaminen ja koulutustarve

Kyselytutkimuksessa vastaajalta kysyttiin näkemyksiä ohjelmistometriikan koulutustarpeista kolmesta näkökulmasta: oma koulutustarve, esimiehen koulutustarve sekä muiden koulutustarve. Vastausvaihtoehdoista koulutuksen osalta laaja koulutustarve käsitti syvällisen kouluttamisen ohjelmistometriikan tuottamisen tai hyödyntämisen näkökulmasta. Peruskoulutus käsitti ohjelmistometriikan yleisesittelyn sekä periaatteet mittaustiedon tuottamisesta sekä hyödyntämisestä.

Ohjelmistometriikan koulutustarve (n = 39)				
Kysymykset (215, 216, 217)	Oma	Esimies	Muut	
Laaja koulutustarve	28 %	5 %	13 %	
Peruskoulutus	41 %	64 %	59 %	
Ei koulutustarvetta	31 %	31 %	28 %	

Taulukko 1: Ohjelmistometriikan koulutustarve

Koulutustarpeisiin liittyvät vastaukset ovat ryhmitelty taulukossa 1 koulutustarpeen näkökulman sekä koulutustarpeen sisällön kesken. Vastanneiden osalta yhteensä 69 % (27 kpl) ilmoitti tarvitsevansa itse koulutusta, yhteensä 69 % (27 kpl) ilmoitti oman esimiehen tarvitsevan koulutusta sekä yhteensä 72 % (28 kpl) ilmoitti muiden tarvitsevan koulutusta. Vastauksista korostuvat ohjelmistometriikan peruskoulutustarve oman itsensä, oman esimiehen sekä muiden osalta. Jokaisesta näkökulmasta tarkasteltuna noin kaksi kolmesta vastaajasta koki ohjelmistometriikan koulutuksen tarpeelliseksi koulutuksen sisällöllisestä tasosta riippumatta. Vastaajista noin kolmannes koki jokaisen näkökulman osalta, että ohjelmistometriikkaan liittyviä koulutustarpeita ei ole.

Kyselytutkimuksessa vastaajalta kysyttiin omaa ohjelmistometriikan osaamista sekä ohjelmistometriikkaan liittyvää koulutustaustaa. Osaamiseen liittyvän kysymyksen avulla oli tarkoituksena selvittää vastaajan omaa ohjelmistometriikan osaamista, tietoisuutta tai tietämättömyyttä. Koulutukseen liittyvän kysymyksen tarkoituksena oli selvittää vastaajan ohjelmistometriikkaa liittyvän koulutuksen tasoa, kuten esimerkiksi sertifiointia tai suoritettua tutkintoa.

Ohjelmistometriikan osaaminen ja koulutustausta (n = 39)				
Kysymykset (211, 212)	Osaaminen	Koulutus		
Osa / Koulutettu	41 %	21 %		
Tietää / Opiskellut	33 %	31 %		
Ei osaa / Ei koulutusta	26 %	48 %		

Taulukko 2: Ohjelmistometriikan osaaminen ja koulutustausta

Ohjelmistometriikan osaamiseen sekä koulutustaustaan liittyvät vastaukset ovat ryhmitelty taulukkoon 2. Kyselyn vastauksista korostui ohjelmistometriikkaan liittyvä osaaminen. Vastanneista yhteensä 74 % (29 kpl) ilmoitti tietävänsä tai osaavansa ohjelmistometriikkaa sekä yhteensä 52 % (20 kpl) oli opiskellut tai saanut ohjelmiestometriikkaan liittyvää koulutusta. Ilmoitetusta osaamisesta huolimatta hieman vajaat puolet ei ollut saanut ohjelmistometriikan koulutusta.

Kyselytutkimuksessa kysyttiin vastaajilta halua osallistua ohjelmistometriikan koulutukseen sekä halua osallistua tuottamaan muille ohjelmiston metriikkaan liittyvää koulutusta. Vastaukset ovat koottu taulukkoon 3 siten, että vastaajan halukkuus osallistua koulutukseen (osallistumishalu) sekä vastaajan halukkuus osallistua tuottamaan muille koulusta (tuottamishalu) ovat tarkasteltavissa yhdessä.

Ohjelmistometriikan koulutukset (n = 39)				
Kysymykset (511, 512)	Osallistumishalu	Tuottamishalu		
Syvällinen koulutus	43 %	15 %		
Yleiskoulutus	31 %	10 %		
Ei ole	26 %	75 %		

Taulukko 3: Osallistumishalukkuus ohjelmistometriikan koulutuksiin

Vastaajista yhteensä 74 % (29 kpl) ilmoitti haluavansa osallistua ohjelmistometriikan koulutuksiin, kun neljännes vastanneista ei halunnut osallistua ohjelmistometriikan koulutukseen. Vastaajista ohjelmistometriikan koulutuksen tuottamiseen sekä muihin järjestelyihin ilmoitti haluavansa osallistua yhteensä 25 % (9 kpl) eli selkeästi.

5.1.2 Ohjelmistometriikan työvälineet

Kyselytutkimuksessa vastaajalta kysyttiin omassa käytössä olevan ohjelmistometriikan työvälinetilannetta sekä mahdollista ohjelmistometriikkaan liittyvään oman työvälineen hankinta- tai päivitystarvetta. Vastausvaihtoehdot luokiteltiin työvälineiden osalta automaattisiin tai manuaalisiin työvälineisiin riippumatta tarpeesta tuottaa tai hyödyntää ohjelmistometriikan mittaustietoa.

Ohjelmistometriikan työväline tilanne ja hankintatarve (n = 39)				
Kysymykset (214, 218)	Nykytila	Hankintatarve		
Automaattinen työväline	26 %	33 %		
Manuaalinen työväline	10 %	5 %		
Ei tarvetta	64 %	62 %		

Taulukko 4: Ohjelmistometriikan työvälinetilanne sekä hankintatarve

Kyselyn vastaukset ohjelmistometriikan työvälineineiden tai hankintatarpeiden osalta ovat koottuna taulukkoon 4. Vastanneiden osalta yhteensä 36 % (14 kpl) ilmoitti, että käytössä on joko automaattinen tai manuaalinen ohjelmistometriikkaan liittyvä työväline. Vastanneista yhteensä 38 % (14 kpl) ilmoitti tarpeesta päivittää käytössä oleva ohjelmistometriikan työväline uudempaan versioon. Kaikista vastauksista korostui näkökulmana, että reilut kaksi kolmasosaa vastaajista ilmoitti olevansa ilman ohjelmistometriikan työvälinettä, työvälineen hankinnan tai päivittämisen olevan tarpeetonta.

Kyselytutkimuksessa kysyttiin vastaajan ohjelmistometriikan ohjeiston saatavuutta ja käyttötarkoitusta. Ohjelmistometriikan ohjeistoon liittyvät vastaukset ovat

koottu taulukkoon 5 siten, että vastaukset ovat ryhmitelty a) laajaan käyttöön (organisaatio), b) rajattuun käyttöön (ratkaisu, palvelu tai projekti) koskevaan ohjeistoon tai c) ohjeiston puuttuminen tai tarpeettomuus.

Ohjelmistometriikan ohjeiston saatavuus (n = 39)				
Kysymykset (419)	Ohjeisto			
Laajaan käyttöön	18 %			
Rajattuun käyttöön	10 %			
Ei ole	72 %			

Taulukko 5: Ohjelmistometriikan ohjeisto

Vastaajista yhteensä 28 % (11 kpl) ilmoitti, että käytettävissä on ohjelmistometriikkaan liittyvä ohjeisto. Vastaajista yhteensä 72 % (28 kpl), ettei ohjelmistometriikan ohjeistoa ei ole tai sen olevan tarpeeton.

5.1.3 Ohjelmistometriikan hyödyntäminen

Kyselytutkimuksessa vastaajalta kysyttiin ohjelmistometriikasta saatavaa hyötyä oman työn näkökulmasta kuin myös tarjouksen sekä sopimuksen osalta. Vastausvaihtoehtoisissa omaan työhön saatava hyöty kohdistui vastaajan rooliin mukaiseen työtehtävään. Ohjelmistometriikasta saatavaa hyötyä tarjouksen tai sopimuksen osalta kysyttiin suoraan vastaajan roolista riippumatta.

Ohjelmistometriikasta saatava hyöty (n = 39)				
Kysymykset (219, 313)	Oma työ	Tarjous, Sopimus		
On hyötyä	41 %	3 %		
Vähäinen hyöty	21 %	18 %		
Ei hyötyä	38 %	79 %		

Taulukko 6: Ohjelmistometriikasta saatava hyöty

Vastaukset ohjelmistometriikasta saatavasta hyödystä ovat koottu taulukkoon 5. Vastanneista yhteensä 62 % (24 kpl) ilmoitti saavansa hyötyä omalle työlle ohjelmistometriikasta. Vastaajista yhteensä 21 % (8 kpl) vastaajista koki ohjelmistometriikasta olevan hyötyä tarjouksen tai sopimuksen kannalta. Vastauksissa on korostunut, ettei 79 % (30 kpl) vastaajista kokenut ohjelmistometriikasta olevan hyötyä tarjouksessa tai sopimuksessa.

Kyselytutkimuksessa vastaajalta kysyttiin ohjelmistometriikkaan liittyvien vaatimuksien sisältymisestä tarjouspyyntöihin. Lisäksi vastaajalta kysyttiin ohjelmistometriikan hyödyntämisen tarpeellisuutta tarjousvalmistelutyössä.

Ohjelmistometriikka tarjouspyynnöissä ja tarjousvalmistelussa (n = 39)				
Kysymykset (311, 312)	Tarjouspyyntö	Tarjousvalmistelu		
Olennainen tekijä / Tulee hyödyntää	13 %	23 %		
On nähnyt / Tulisi hyödyntää	18 %	28 %		
Ei ole / Ei hyötyä	69 %	49 %		

Taulukko 7: Ohjelmistometriikka tarjouspyynnöissä ja tarjousvalmistelussa

Vastaukset ohjelmistometriikasta tarjouspyynnön sekä tarjousvalmistelun osalta ovat koottu taulukkoon 7. Vastanneista yhteensä 31 % (12 kpl) oli nähnyt tarjouspyynnössä ohjelmistometriikkaan liittyviä vaatimuksia. Yhteensä 51 % (19 kpl) ilmoitti, että ohjelmistometriikkaa on tarpeen hyödyntää osana tarjousvalmistelutyötä. Vastanneista 69 % (26 kpl) ei ollut nähnyt tarjouspyynnöissä vaatimuksia liittyen ohjelmistometriikkaan sekä yhteensä 49 % (19 kpl) koki, ettei ohjelmistometriikasta ole hyötyä osana tarjousvalmistelutyötä.

Kyselytutkimuksen vastaajalta kysyttiin ohjelmistometriikan hyödyntämistä estimoinnissa eli työmäärä-, aikataulu- ja kustannusarvioiden tuottamisen tukena.

Ohjelmistometriikka tarjousvalmistelussa sekä estimoinnissa (n = 39)				
Kysymykset (312, 313)	Tarjousvalmistelu	Estimointi		
Tulee hyödyntää / Hyödynnän Usein	23 %	18 %		
Tulisi hyödyntää / Hyödynnän harvoin	28 %	23 %		
Ei hyötyä / En hyödynnä	49 %	59 %		

Taulukko 8: Ohjelmistometriikka tarjousvalmistelussa sekä estimoinnissa

Kyselyn vastaukset tarjousvalmistelun sekä estimoinnin osalta ovat koottu taulukkoon 8. Vastaajista yhteensä 51 % (19 kpl) koki, että ohjelmistometriikkaa olisi syytä hyödyntää osana tarjousvalmistelua. Vastaajista yhteensä 41 % (16 kpl) ilmoitti hyödyntävänsä ohjelmistometriikkaa toimitukseen liittyvässä arviointityössä. Vastaajista 49 % (19 kpl) ei koe hyötyä ohjelmistometriikasta tarjousvalmistelussa eikä 59 % (23 kpl) hyödynnä ohjelmistometriikkaa arvioiden laatimisessa.

Kyseytutkimuksessa kysyttiin vastaajan ohjelmistometriikan hyödyntämisestä ennen työtä, työn tekemisen aikana sekä työn tekemisen jälkeen. Kyselyn vastaukset ovat koottu taulukkoon 9 ohjelmistometriikan yleisen hyödyntäminen

(Yleisesti-sarake) sekä ohjelmistometriikan hyödyntämisestä xshoren (alihankkijat, near shore, offshore, bleded shore) seurannan yhteydessä (xshore-sarake).

Ohjelmistometriikkaan hyödyntäminen ennen työtä, työaikana ja työn jälkeen sekä xshore-toiminnassa (n = 39)				
Kysymykset (415, 416)	Yleisesti	xshore		
Hyödynnän	10 %	0 %		
Hyödynnän ajoittain	41 %	13 %		
En hyödynnä	49 %	87 %		

Taulukko 9: Ohjelmistometriikan hyödyntäminen työssä ja xshoressa

Vastaajista yhteensä 51 % (19 kpl) hyödyntää ajoittain tai usein ohjelmistometriikkaan ennen työn tekemistä, työn tekemisen aikana sekä työn tekemisen jälkeen. Vastaavasti 13 % (5 kpl) hyödyntää ohjelmistometriikkaan xshoren toiminnan yhteydessä. Vastauksissa korostuu, että puolet vastajista ei hyödynnä yleisesti ohjelmistometriikkaa eikä suurin osa hyödynnä lainkaan ohjelmistometriikkaan xshoren toimituksien yhteydessä. Tuloksien osalta tulee huomioida, ettei kyselytutkimuksesta selviä, kuinka vastaajista oli mukana xshore-toiminnassa.

5.1.4 Ohjelmistometriikan käyttöönotto

Kyselytutkimuksen vastaajalta kysyttiin yleistä työn mittamiseen liittyvän kulttuurin nykytilaa. Kyselyssä tarkasteltiin mittaamista sekä asetettujen raja-arvojen näkökulmista nykytilaa ennen työn tekemistä (asetan-sarake), työn tekemisen aikana (seuraan), työn tulosten todentamista (todennan) sekä tarvittaessa työn tulosten korjaamista (korjaan).

Mittaamiskulttuurin nykytila ja raja-arvojen hyödyntäminen (n = 39)				
Kysymykset (411, 412, 414, 413)	Asetan	Seuraan	Todennan	Korjaan
Kyllä	33 %	13 %	38 %	31 %
Ajoittain	46 %	38 %	36 %	46 %
En	21 %	49 %	26 %	23 %

Taulukko 10: Mittaamiskulttuurin nykytila

Mittaamisen sekä raja-arvoihin perustuvan toiminnan nykytila on koostettu taulukoon 10. Vastaajista yhteensä 79 % (30 kpl) asetti tehtävälle etukäteen raja-arvot. Vastaajista yhteensä 51 % (19 kpl) hyödynsi mittaamista sekä asetettuja raja-arvoja seurannan apuvälineenä. Vastaajista työn tuloksia todensi asetettujen raja-arvojen avulla yhteensä 74 % (28 kpl) sekä 77 % (30 kpl) vastaajista myös korjasi tarvittaessa tehdyn työn tuloksia mittauksen perusteella.

Kyselututkimuksessa vastaajalta kysyttiin ohjelmistometriikan tietojen keräämisestä myöhempää hyödyntämistä varten sekä kerätyn ohjelmistometriikan hyödyntämistä. Vastaukset ovat koottu taulukkoon 11 ohjelmiston metriikan keräämisen (Kerään-sarake) sekä kerätyn ohjelmiston metriikan hyödyntämisen (Hyödynnän-sarake) osalta.

Kerään ohjelmistometriikkaa ja hyödynnän tai opin aiemmin kerätystä ohjelmistometriikasta (n = 39)				
Kysymykset (417, 418)	Kerään	Hyödynnän		
Kyllä	8 %	28 %		
Ajoittain	33 %	28 %		
En	59 %	44 %		

Taulukko 11: Ohjelmistometriikan kerääminen ja tiedon hyödyntäminen

Vastaajista yhteensä 41 % (16 kpl) kerää ohjelmistometriikka tietoa. Yhteensä 56 % (21 kpl) vastaajista hyödyntää kerättyä ohjelmistometriikkaa tai pyrkii oppimaan siitä. Vastauksissa korostuu, että 59 % (23 kpl) ei kerää ohjelmistometriikkaa eikä 44 % (17 kpl) hyödynnä kerättyä ohjelmistometriikkaa tai pyri oppimaan siitä.

Kyselytutkimuksessa kysyttiin vastaajilta esimiehen antamaa tukea sekä kannustusta vastaajan osaamisen kehittämiseksi. Vastaukset ovat koottu taulukkoon 12.

Esimiehen tuki ja kannustus (n = 39)				
Kysymykset (513)	Esimiehen tuki			
Saa tukea pääsääntöisesti aina	5 %			
Saa tukea ajoittain	77 %			
Ei saa tukea	18 %			

Taulukko 12: Esimiehen antama tuki ja kannustus

Vastanneista yhteensä 82 % (32 kpl) ilmoitti saavansa esimieheltä tukea sekä kannusta kehittää itseään parempien työsuorituksien saavuttamiseksi. 18 % (7 kpl) vastanneista koki, ettei esimies tue tai kannusta kehittymään.

Kyselytutkimukseen vastanneilta kysyttiin näkemystä ohjelmistometriikan käytönoton suurimmista esteistä. Vastausvaihtoehdot ovat ryhmitelty taulukossa 13 toimintaedellytyksiin, tietotaitoon tai ei estettä.

Ohjelmistometriikan käyttöönoton suurimmat esteet (n = 39)				
Kysymykset (514)	Suurimmat esteet			
Toimintaedellytys	23 %			
Tietotaito	51 %			
Ei ole	26 %			

Taulukko 13: Ohjelmistometriikan käyttöönoton esteet

Vastaajista yhteensä 74 % (28 kpl) ilmoitti, että ohjelmistometriikan käyttöön oton keskeisinä esteinä ovat ohjelmistometriikkaan liittyvä tietotaito sekä yleinen toimintaedellytys. Vastaajien mukaan suurin ohjelmistometriikan käyttöönoton estävä tekijä on tietotaito. Vastaajista 26 % (10 kpl) koki, ettei ohjelmistometriikan käyttöönotolle ole estettä.

Kyselytutkimukseen vastanneilta kysyttiin suositusta ohjelmistometriikan kohderyhmästä. Vastausvaihtoehdot ryhmiteltiin taulukossa 14 toimitukselle (projekti-ryhmä), kaikille tai ei suosittele lainkaan.

Suosittelee ohjelmistometriikkaa (n=39)				
Kysymykset (514)	Suositus			
Hallinnolle ja kaikille muille	28 %			
Toimitukselle	57 %			
En suosittele	15 %			

Taulukko 14: Suositus ohjelmistometriikan kohderyhmästä

Vastanneista yhteensä 85 % (33 kpl) suosittelee yleisesti ohjelmistometriikan käyttöä. Kaikkiaan 57 % (22 kpl) suosittelee ohjelmistometriikan käyttöä ohjelmistoprojektin toimituksessa. Lisäksi 28 % (10 kpl) vastaajista suosittelee ohjelmistometriikkaa hallinnolle ja myös kaikille muille. Vastajista 15 % (5 kpl) ei suositellut ohjelmistometriikan hyödyntämistä.

5.2 Kyselytutkimuksen tulosten analysointi

Kohdeyrityksessä on tuhansia työntekijöitä. Kyselytutkimuksen kohdennettiin vain pienelle joukolle vastaajia ”fiilisbarometrin” tuottamiseksi kohdeyrityksen yhdessä liiketoimintayksikössä. Tämä osaltaan tarkoittaa, että kyselytutkimuksen lopputulosta ei voida yleistää koskemaan kohdeyritystä.

Kyselytutkimuksen kysymykset ohjelmistometriikan hyödyntämisestä olivat esitetty laveana kysymyksenä eikä niihin ollut määritelty tarkkaa hyödyntämistapaa.

Tällöin vastaajan subjektiivinen näkemys sekä aihepiirin tuntemus on mahdollisesti vaikuttanut tuloksiin. Kyselytulosten osalta tulee myös huomioda, että kaksi kolmasosaa vastaajista edustivat ohjelmistototeutusryhmän eri rooleja, mistä syystä lopputulokset saattavat painottua projektitoimituksen sisäiseen näkökulmaan.

Kyselytutkimuksen vastauksien perusteella kohdeyrityksessä oli vaihteleva käytäntö ohjelmistometriikan hyödyntämisen osalta. Syy tähän löytynee kohdeyrityksen nykyisestä tekemisen mallista, joka ei edellyttänyt ohjelmistometriikan hyödyntämistä. Ohjelmistometriikan tietämyksen sekä osaamisen kasvattaminen voi luoda uusia mahdollisuuksia työn tulosten laadun parantamiseksi sekä siten myös parantaa mahdollisuuksia saavuttaa parempia taloudellisia tuloksia.

Kyselytutkimuksen mukaan ohjelmistometriikasta yli puolet vastaajista saa hyötyä omalle työlle. Tämän perusteella ohjelmistometriikan systemaattisen hyödyntämisen laajentaminen saattaisi olla yksi varteen otettava vaihtoehto, koska pohjalla on positiivinen asenne sekä toiseksi nykyisellään yli puolet vastaajista hyödyntää jo ohjelmistometriikkaa.

Vastauksien perusteella voidaan todeta suurimman osan suhtautuvat positiivisesti koulutukseen, mikä luo hyvän vastaanottavaisen pohjan ohjelmistometriikan käytön laajentamiselle, koulutuksille sekä käyttöönotolle.

Ohjelmistometriikan hyödyntämisen määrittely, suunnittelu sekä toimintajärjestelmän päivittäminen tarvittavilta osin edellyttäneen suurehkoa työpanosta. Ohjelmistometriikan soveltamisen määrämuotoistaminen sekä käytön laajentaminen edellyttää ohjelmistometriikkaan liittyvien koulutuksien suunnittelua, järjestämistä sekä toteutusta. Suurella ihmismäärällä tämä aiheuttavat merkittäviä kustannuksia sekä lyhyellä aikajänteellä liikevaihdon menetyksiä menetetyn laskutettavan työajan muodossa.

5.2.1 Ohjelmistometriikan osaaminen ja koulutustarve

Kyselytutkimuksen ohjelmistometriikan koulutustarpeiden (taulukko 1) kartoituksen osalta korostui tarve ohjelmistometriikan yleiskoulutukselle vastaajan itsensä, oman esimiehen kuin myös työkaveriden osalta. Vastanneiden keskeisen oma

yleiskoulutustarve oli hieman matalampi kuin oli näkemys esimiehen sekä työkalvereiden koulutustarpeesta. Tämä johtunee siitä, että suurin osa vastaajista edusti projektiryhmän jäseniä, joista osalla oli entuudestaan ohjelmistometriikan osaamista ja käytettävissä myös työväline.

Kyselytutkimuksien vastauksien perusteella oli tulkittavissa, ettei ohjelmistometriikan osaaminen ollut riittävällä tasolla esimiehillä tai muilla työstä vastaavilla. Tämän lienee käytännössä myös perusteltu tulkinta, koska kohdeyrityksessä esimiesten sekä työstä vastaavien rooliin kuuluvat pääasiassa projektien osalta toimintaedellytysten järjestäminen, työmäärien, aikataulujen sekä talousasioiden seuranta.

Oman ohjelmistometriikan yleiskoulutustarpeen sijaan kolmannes vastaajista koki tarvetta laajammalle ohjelmistometriikan koulutukselle sisältäen työvälineisiin liittyvän koulutuksen. Noin kolmannes vastaajista ei kokenut tarvetta koulutukseen itsensä, esimiehensä tai muiden osalta. Tähän syynä saattavat olla tietämättömyys tai riittävä ohjelmistometriikan osaaminen entuudestaan.

Kohdeyrityksessä on suositeltavaa järjestää ohjelmistometriikan yleiskoulutus henkilölle roolista riipumatta, henkilön esimiehelle sekä työkalvereille. Syvällisempi jatkokoulutus sisältäen työvälineiden soveltamisen on syytä järjestää rajatulle kohderyhmälle.

Ohjelmistometriikan osaamisen sekä ohjelmistometriikan koulutustaustan (taulukko 2) osalta kyselytutkimus osoittaa, että ohjelmistometriikka asiana tiedetään ja osataan. Koulutustaustan näkökulmasta tarkasteluna vain viidennes on suorittanut kuitenkin ohjelmistometriikkaan liittyvän tutkinnon tai sertifikaatin. Käytännössä tämä tarkoittanee, että vastaajat olivat itse perehtyneet ohjelmistometriikkaan tai ovat oppineet käytännön työn yhteydessä. Tämä johtanee siihen, että kohdeyrityksessä ohjelmistometriikan osaaminen saattaa vaihdella ja yhteinen osaamispohja puuttuu.

Vastaajilla oli halu osallistua yleiseen sekä syvälliseen ohjelmistometriikan koulutukseen (taulukko 3). Vastaajilla ei sen sijaan ollut ollut halua järjestää muille ohjelmistometriikan koulutusta. Tästä johtuen on suositeltavaa harkita yleisen ohjelmistometriikan koulutuksen ostamista kohdeyrityksen ulkopuolelta. Koh-

deyriksen oma kehittämispanos tulisi keskittää mielummin ohjelmistometriikan soveltamismallin sovittamiseen osaksi kohdeyriksen toimintajärjestelmää kuin perusasioiden järjestämiseen.

Kohdeyriksessä on suositeltavaa järjestää yleinen koulutus ohjelmistometriikan perusteista, jotta kohdeyrikseseen saadaan luotua yhteinen ohjelmistometriikan osaamispohja, näkemys sekä kieli. Tämän lisäksi koulutukseen tulee sisällyttää ohjelmistometriikan käytännön hyödyntäminen osana kohdeyriksen toimintamallia sekä prosesseja. Ohjelmistometriikan soveltaminen osana kohdeyriksen arkipäivää edellyttää ohjelmistometriikan soveltamiseen liittyvien tehtävien sekä vastuiden upottamista osaksi kohdeyriksen toimintajärjestelmää. Kohdeyriksen toimintajärjestelmän osalta tehtävät sekä vastuut tulisi päivittää roolien näkökulmasta.

5.2.2 Ohjelmistometriikan työvälineet

Kyselytutkimus osoitti ohjelmistometriikan työvälineiden (taulukko 4) osalta, että reilulla kolmanneksella oli käytettävissä ohjelmistometriikkaan liittyvä työväline. Lisäksi kolmanneksella oli tarve hankkia tai päivittää ohjelmistometriikan työväline.

Ohjelmistometriikan käytön laajentaminen sekä määrämuotoistaminen tulee edellyttämään niin henkilökohtaiseen kuin yhteiseen käyttöön liittyviä työvälinehankintoja sekä työvälineisiin liittyviä koulutuksia. Näistä aiheutuu vastaavasti lisää hankinta- sekä käyttöönottokustannuksia. Tässä yhteydessä ohjelmistometriikan työvälineitä ei tule hankkia laajempaan käyttöön ennen kuin kohdeyriksessä on selvillä yhteinen ohjelmistometriikan soveltamistapa. Toiseksi kohdeyriksessä tuotetaan ohjelmistoratkaisuja eri ohjelmistoteknologioilla sekä niihin liittyvillä työvälineillä. Tämä tarkoittaa käytännössä sitä, että ohjelmistokehittäjien tai testaajien osalta ei ole mahdollista valita yhtä yksittäistä ohjelmistometriikan työvälinettä kohdeyriksen tarpeisiin. Johtamisen sekä muun soveltamisen näkökulmasta kohdeyrikselle olisi mahdollista hankkia yhteinen ohjelmistoteknologia riippumaton työväline mittaustiedon ylläpitoon, raportointiin sekä hyödyntämi-

seen. Ennen välineiden laajamittaista hankintaa on syytä kehittää toimintamallia siten, että ohjelmistometriikan soveltaminen uppoaa osaksi normaalia toimintaa.

Ohjelmistometriikan työvälineiden osalta hankintatarpeet saattavat kasvaa kolmannesta korkeammiksi mahdollisten koulutuksien, yleisen tietämyksen ja osaamisen kasvaessa. Tämän takia työvälinehankintoihin liittyvässä budjetoinnissa tulee varautua kasvaviin työvälineiden lisenssien hankintatarpeisiin.

Kohdeyrityksen näkökulmasta lähitulevaisuudessa on suositeltavaa hankkia uusia tai päivittää käytössä olevia ohjelmistometriikan työvälineitä vain tilanteen sekä tarpeen mukaan. Laajemmat mahdolliset työvälinehankinnat on syytä ajoittaa toimintamallin kehittämisen yhteyteen myöhemmin tulevaisuudessa.

5.2.3 Ohjelmistometriikan hyödyntäminen

Kyselytutkimuksen mukaan ohjelmistometriikasta ei ole hyötyä tarjouksessa tai sopimuksessa (taulukko 7). Myynnillisesti ohjelmistometriikan osalta oli havaittavissa se, ettei ohjelmistometriikkaan liittyviä vaatimuksia ollut juurikaan esiintynyt tarjouspyynnöissä. Kuitenkin puolet vastaajista esitti, että ohjelmistometriikka tulisi hyödyntää tarjousvalmistelussa.

Kyselytutkimuksen perusteella kaksi kolmesta ei hyödynnä ohjelmistometriikkaa työmäärien, aikataulujen tai kustannusten arvioinnissa (taulukko 8). Ohjelmistometriikan hyödyntäminen arvioiden laatimisessa saattaisi olla kaupallisesti hyödyllistä, mistä syystä tulisi selvittää tarkemmin ohjelmistometriikan hyödyntämisen osana kaupallisia työvaiheita.

Ohjelmistometriikan kaupallisen hyödyntämisen lopputuloksen osalta tulee huomioda, että kyselyyn vastanneiden muiden kuin myyntityötä tekevien osuus oli suuri. Tästä syystä, kohdeyrityksen tulisi käsitellä asiana ohjelmistometriikan hyödyntäminen osana tarjousvalmistelua sekä sopimuksia.

Kohdeyrityksessä yli puolet koki saavan hyötyä omaan työhönsä, mutta vastaavasti hyötyä ei koeta saatavan tarjouksien tai sopimuksien osalta (taulukko 6). Tältä pohjalta on suositeltavaa aloittaa ohjelmistometriikan määrämuotoistaminen ohjelmistotoimitusprojektien sekä ylläpidon osalta.

Kohdeyrityksessä ohjelmistometriikkaa hyödynnettiin ennen työn tekemistä, työn tekemisen aikana sekä työn tekemisen jälkeen (taulukko 9). Noin puolet vastanneista ei hyödyntänyt ohjelmistometriikkaan työn tekemisen tukena. Lisäksi ohjelmistometriikkaan liittyvää ohjeisto puuttui tai sitä oli vähän. (taulukko 5).

Kohdeyrityksessä vastanneiden keskuudessa ei sovelleta ohjelmistometriikkaa alihankkijoiden, near shoren, offshoren tai blended shoren kanssa (taulukko 9). Tässä kohdin ohjelmistometriikan soveltaminen saattaisi auttaa merkittävästi alihankkijoiden tai xshore tyyppisen toiminnan ohjelmistoteknisen laadun tavoiteasettelu, seurannan kuin myös todentamisen osalta, koska ohjelmistometriikan mittareiden avulla olisi saada objektiivista tietoa seurannan, hyväksynnän sekä päätöksenteon tueksi.

Kohdeyrityksessä on suositeltavaa tutkia erikseen ohjelmistoprojektien osalta ohjelmistometriikan hyödyntäminen. Erityisesti kohdeyrityksen kannalta tulisi kiinnittää huomiota ohjelmistometriikan hyödyntämisessä xshore-toiminnassa.

5.2.4 Ohjelmistometriikan käyttöönotto

Suurimmat esteet ohjelmistometriikan käyttöönotolle löytyivät tietotaidosta sekä toimintaedellytyksistä. Koulutuksien avulla on mahdollista lisätä tietotaitoa sekä näkemyksistä, jolloin mahdollisesti myös ohjelmistometriikan käyttöönoton esteet vähenevät entisestään.

Kohdeyrityksessä esimiehet tukivat sekä kannustivat alaisiaan kehittämään omaan osaamistaan (taulukko 12). Tämä loi hyvän lähtökohdan ohjelmistometriikan määramuotoistamiselle sekä muutokselle.

Kohdeyrityksessä oli toimintajärjestelmä, mitä päivittämällä sekä täydentämällä olisi mahdollista saada kohtuullisella työllä tarvittava toimintamalli sekä ohjeisto ohjelmistometriikan osalta kaikkien saataville.

Kohdeyrityksessä oli käytössä ohjelmistometriikkaa tukevia työvälineitä. Työvälinetilanteen näkökulmasta ohjelmistometriikan käyttöönoton aloittaminen pilotin muodossa onnistunee parhaimmillaan ilman työvälinehankintoja.

Kyselytutkimuksen mukaan suhtautuminen ohjelmistometriikkaan oli myönteinen, koska valtaosa vastaajista suositteli ohjelmistometriikkaa. Tämä loi osaltaan vastaanottavaisen ilmapiirin ohjelmistometriikan käyttöönotolle tai hyödyntämisen määrämuotoistamiseksi.

Tulosten perusteella on mahdollista todeta, että kohdeyrityksessä mittaamiskulttuurin perustuva toiminta oli vaihtelevaa. Mittaamiseen sekä raja-arvoihin perustuva toiminta on keskenään tasapainossa asettamisen, todentamisen sekä korjaamisen osalta. Seurannan tasoa poikkesi muiden osa-alueiden tasosta.

Kyselytutkimuksen perusteella kohdeyrityksessä mittaamiskulttuuri oli hyvissä kantimissa (taulukko 10). Mittaamisen näkökulmasta tavoitteiden (raja-arvojen) asettaminen, seuranta, todentaminen sekä korjaavien toimenpiteiden suorittaminen olivat osa yrityksen arkipäivää. Tämä luo hyvän pohjan ohjelmistometriikan käytäntöjen kehittämislle sekä soveltamiselle kohdeyrityksessä.

Kohdeyrityksessä ohjelmistometriikka tulee kytkeä eri roolien näkökulmasta tekemiseen kokonaisuuten soveltuvalla tavalla. Esimerkiksi myynnillisesti ohjelmistometriikan soveltaminen osanan ratkaisun luontia sekä siihen liittyvien arvioiden tekemistä, projektipäällikön näkökulmasta osana työnohjausta, ohjelmistokehittäjän tai -ylläpitäjän näkökulmasta osana ohjelmiston teknistä toteutusta, testaajan näkökulmasta asioiden todentamisen kannalta, projektiryhmän ohjausryhmän jäsenen sekä esimiehen näkökulmasta sekä ulkoisesta asiakkaan näkökulmasta.

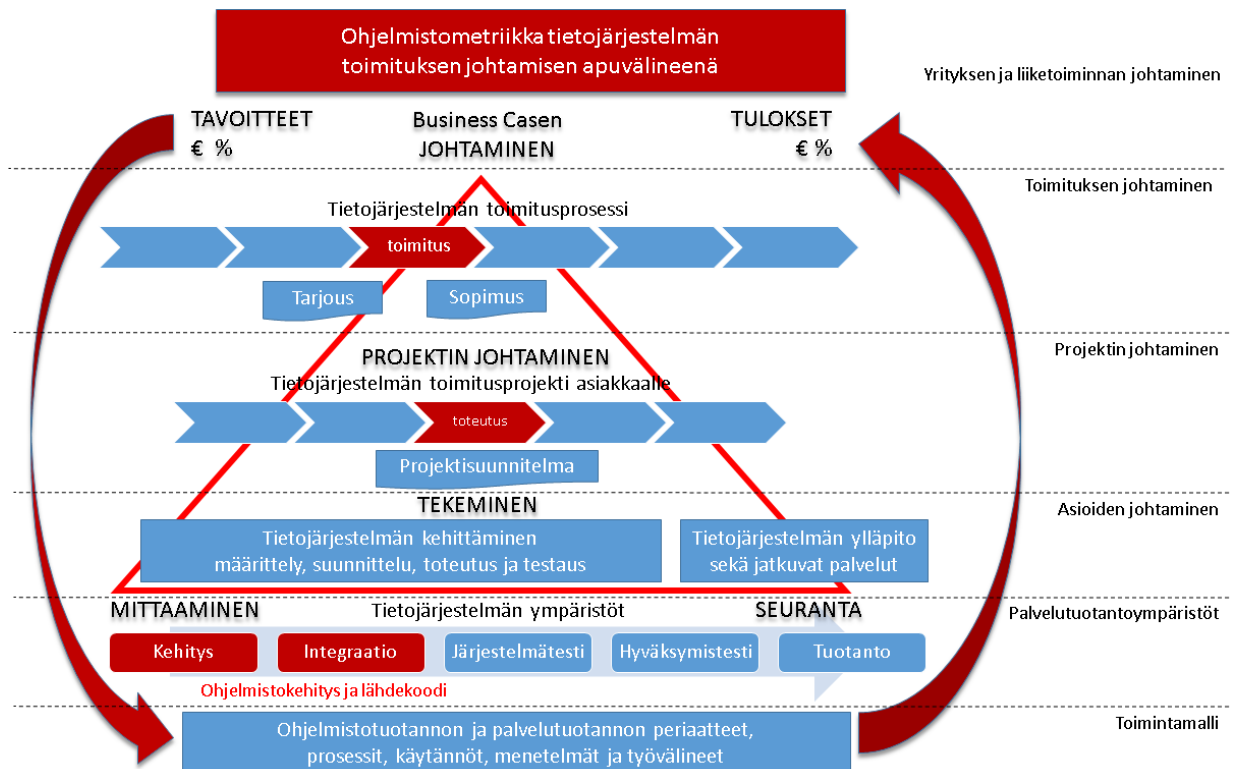
Kohdeyrityksessä on suositeltavaa valita soveltuvat ohjelmistometriikan mittarit eri näkökulmiin sekä liittää ne osaksi katselmointeja sekä seurantaraportteja asioiden johtamisen sekä päätöksen teon tueksi.

Kyselytutkimuksen perusteella kohdeyrityksessä mittaamiskulttuuri oli hyvissä kantimissa (taulukko 11). Mittaamisen näkökulmasta tavoitteiden (raja-arvojen) asettaminen, seuranta, todentaminen sekä korjaavien toimenpiteiden suorittamine kaikkien saataville n olivat osa kohdeyrityksen arkipäivää. Tämä loi hyvän pohjan ohjelmistometriikan käytäntöjen kehittämislle sekä soveltamiselle kohdeyrityksessä.

Kohdeyrityksessä on suositeltavaa valita soveltuvat ohjelmistometriikan mittarit eri näkökulmiin sekä liittää ne osaksi katselmointeja sekä seurantaraportteja asioiden johtamisen sekä päätöksen teon tueksi.

5.3 Tutkimuksen lopputulos ja toimenpidesuositus kohdeyritykselle

Tutkimuksen aikana loin ohjelmistometriikan teorian sekä kohdeyrityksen toimintajärjestelmän yksinkertaistettujen mallien pohjalta uuden kokonaiskuvan (kuva 19) kartaksi ohjelmistometriikan sijoittumista osaksi kohdeyrityksen toimintaa. Kokonaiskuvan ydinajatus on, että ohjelmistometriikan hyödyntäminen tulee kytkeä toiminnan eri tasoille niin työn tekemisen kuin myös johtamisen osalta.



Kuva 17: Ohjelmistometriikan sijoittuminen kohdeyrityksen toiminnassa

Kokonaiskuvan (kuva 19) oikeassa reunassa katkoviivoilla on esitetty toiminnan tasoja. Kuvassa liiketoiminnan johtamisen alle sijoittuu tietojärjestelmätoimituksen business case, joka realisoituu tietojärjestelmätoimitusprosessina. Business case on liiketoiminnallinen kokonaisuus, jonka tulee olla kannattava. Business case voi olla yksittäinen projekti, jonka läpivienti pohjautuu tarjoukseen, sopimukseen sekä projektisuunnitelmaan. Tietojärjestelmätoimitusprojektiin sisältyy ohjelmiston määrittely, suunnittelu, toteutus ja testaus. Tietojärjestelmän ohjel-

mistoteknistä toteutusta sekä testaamista varten tulee olla työympäristöjä, kuten kehitys- ja integraatiotestiympäristöt, joihin ohjelmistometriikan soveltaminen sijoittuu pääasiassa. Edellä kuvattu kokonaisuus perustuu uudella tavalla esitettyinä kohdeyrityksen toimintajärjestelmään, joka kuvaa prosessit, toimintamallit, menetelmät sekä työohjeet.

Kokonaiskuvassa (kuva 19) johtaminen, tavoitteiden asettaminen, tekeminen, mittaaminen ja seuranta leikkaavat tekemisen läpi kaikilla tasoilla. Ohjelmistometriikalle tulee vastaavasti asettaa tavoitteet, mitä on mahdollista mitata sekä seurata osana tietojärjestelmän kehittämistä ja ylläpitoa. Kokonaiskuvassa on korostettu ohjelmistokehityksen ja lähdekoodin sijoittuminen kokonaiskuvassa konkreettisen tekemisen osalta. Mahdollisen jatkokehitysprojektin osalta uutta kokonaiskuvaa on mahdollista hyödyntää esimerkiksi apuvälineenä suunniteltaessa ohjelmistometriikan kytkemistä osaksi muuta toimintamallia.

Tutkimusongelma oli määritelty seuraavasti:

Mitä hyötyä on ohjelmistometriikasta asiakaskohtaisesti räätälöitävän tietojärjestelmän kehittämisen ja ylläpidon liiketoiminnan johtamiselle kohdeyritykselle?

Hyödyt kohdeyritykselle ohjelmistometriikan soveltamisesta asiakaskohtaisesti räätälöitävän tietojärjestelmän kehittämisen ja ylläpidon liiketoiminnan johtamisen osalta ovat ohjelmistoteknisen toteutuksen teknisen laadun parantaminen sekä tietojärjestelmän toimituksen ja ylläpidon aikaisten laatu- ja kustannusten minimointi. Ohjelmistometriikkaan liittyvän systemaattisen soveltamisen kehittämisellä kohdeyritys voi parantaa tietojärjestelmän toimituksen suunnitellun aikataulun sekä kustannuksien toteuttamista. Ohjelmistometriikan hyödyntämisen avulla kohdeyritys lisää ohjelmistotekniseen laadun objektiivista mittaamista sekä mahdollisten laatu- ja kustannuspoikkeamien osalta proaktiivista toimintaa, mikä osaltaan tukee operatiivista johtamista. Tätä kautta ohjelmistometriikan soveltaminen vaikuttaa myös kohdeyrityksen keskeisiin johtamisen osa-alueisiin asiakastyytyväisyyden, henkilöstötyytyväisyyden sekä myös liiketoiminnallisen tuloksen osalta.

- 1. Miten kohdeyrityksessä ohjelmistometriikkaa tulisi hyödyntää osana tietojärjestelmän myyntiä, kehittämistä ja ylläpitoa?*

Asiakaskohtaisesti räätälöitävän tietojärjestelmän myynnin näkökulmasta ohjelmistometriikalla ei ollut suoraa hyötyä. Tarjousvalmistelussa ohjelmistometriikan soveltaminen teknisen toteutuksen laajuuden arvioinnissa olisi hyödyllinen apuväline työmäärä-, aikataulu- sekä kustannusarvioiden tuottamiseksi. Asiana ohjelmistometriikan soveltaminen kaupallisista näkökulmista tulisi tutkia erikseen omana jatkotutkimuksena.

Tietojärjestelmän kehittämisen näkökulmasta ohjelmistometriikasta on hyötyä teknisen laadun parantamisessa sekä laatukustannusten minimoimisessa, koska ohjelmistometriikan avulla on mahdollista tuottaa kustannustehokkaasti sekä nopeasti objektiivista tilannetietoa toimituksen johtamisen tueksi. Toiseksi ohjelmistometriikan soveltaminen systemaattisesti ennen työn tekemistä, tekemisen aikana sekä jälkeen lisää ohjelmistoteknisen toteutuksen osalta mahdollisuuksia parantaa teknistä laatua, jolloin teknisen toteutuksen aikana tai sen jälkeen seuraavissa toimituksen vaiheissa on mahdollista saavuttaa kustannussäästöjä.

Tietojärjestelmän ylläpidon näkökulmasta ohjelmistometriikan soveltaminen tietojärjestelmän teknisen toteutuksen aikana parantaa teknistä laatua sekä selkeyttää myös ohjelmiston sisäistä rakennetta ja monimutkaisuutta. Tällöin ohjelmiston ylläpito on myös helpompaa, tehokkaampaa sekä nopeampaa, jolloin vastaavasti on mahdollista säästää aikaa sekä kustannuksia. Edellä mainitut parantavat osaltaan ylläpitovaiheen asiakastyytyväisyyttä sekä liiketoiminnallista kannattavuutta.

2. Mitkä kohdeyrityksen asiat mahdollistavat järjestelmällisen ohjelmistometriikan käyttöönoton ja soveltamisen osana arkipäivän työtä?

Kohdeyrityksessä oli olemassa mittaamiskulttuuri sekä hyvä henkinen vastaanotokyky ryhtyä soveltamaan ohjelmistometriikkaa nykyistä määrämuotoisemmin. Lisäksi ohjelmistometriikka oli jossain määrin entuudestaan tuttu ja osalla vastaajia oli käytössään ohjelmistometriikkaan liittyviä työvälineitä. Kohdeyrityksen toimintamalli olivat kuvattu sekä ohjeistettu toimintajärjestelmässä prosessien, käytäntöjen sekä menetelmien osalta, joihin ohjelmistometriikkaan liittyvät asiat tulisivat täydentää. Erityisen tärkeä näkökulma oli, että esimiehet kannustivat sekä tukivat alaisiaan kehittämään omaa osaamistaan. Lisäksi vastaajat kokivat ohjel-

mistometriikan koulutuksen tarpeelliseksi. Samalla löytyi myös halukkuutta osallistua koulutuksiin.

3. Mitkä kohdeyritykset asiat jarruttavat ohjelmistometriikan käyttöönottoa ja hyödyntämistä?

Suurimmat esteet ohjelmistometriikan määrämuotoisemmalle käyttöönotolle kohdeyrityksessä ovat aihepiirin liittyvä tietotaito, osaaminen sekä näkemykset asian soveltamisesta. Kohdeyrityksen nykyinen toimintajärjestelmä ei edellytä ohjelmistometriikan soveltamista. Ohjelmiston tekniseen laattuu liittyvät selkeät objektiivisest mittarit puuttuvat kohdeyrityksen projektin johtamisen tai tietojärjestelmän toimituksen johtamisesta. Yhteisten käytäntöjen puuttuessa ohjelmistometriikan soveltaminen on hajanaista eikä ohjelmistometriikan soveltamista osata vaatia.

4. Miten kohdeyrityksen ohjelmistotuotannon seurannan ja mittaamisen yhteyttä tulisi parantaa osaksi liiketoiminnan mittaamista ja seurantaa?

Ohjelmistometriikka tulisi kohdeyrityksessä sitoa osaksi tietojärjestelmän toimituksen sekä myös projektien johtamisesta siten, että ohjelmistometriikka teknisen laadun osalta nostetaan sovellettavaksi osaksi muuta toimituksen tai projektin johtamisen mittaristoa. Käytännössä tulisi kehittää operatiivisen sekä projektin johtamiskäytäntöjä, valita soveltuvat mittarit, mahdollistaa ohjelmiston teknisen laadun mittaaminen sekä soveltaminen kohdeyrityksen toiminnan eri tasoilla. Ohjelmistometriikan soveltamisessa tulisi huomioida soveltaminen yksittäisen business casen kuin myös muun liiketoiminnallisen kokonaisuuden näkökulmasta.

Kohdeyrityksessä tulee määrämuotoistaa sekä laajentaa ohjelmistometriikan hyödyntämistä osanan räätälöityjen tietojärjestelmien toimittamista. Tämän saavuttamiseksi suosittelen oman erillisen ohjelmistometriikan hyödyntämiseen liittyvän kehityshankkeen suunnittelun ja toteutuksen läpivientiä. Tämän lisäksi suosittelen kehityshankkeen lopputulosten vaiheittaista jalkautusta osaksi organisaation normaalia toimintaa johtamisen ja ohjelmistoprojektienläpiviennin tukemiseksi sekä ohjelmistojen teknisen laadun parantamiseksi.

Ohjelmistometriikan hyödyntämisen määrämuotoistamiseen liittyvässä kehityshankkeessa tulee täsmentää kohdeyrityksen olemassa olevaa toimintamallia, linjauksia sekä prosesseja käsittämään myös ohjelmistometriikka. Tämä edellyttää ohjelmistometriikan soveltamistavan määrittelyä sekä suunnittelua. Lisäksi kohdeyrityksen toimintajärjestelmän sisältämät prosessikuvaukset, ohjeet sekä näihin liittyvät pohjadokumentit tulee päivittää.

Kohdeyrityksessä tulee järjestää ohjelmistometriikan koulutus roolikohtaisesti, jotta eri roolien näkökulmasta asiasisällön painotukset voivat vaihdella ohjelmistometriikan tuottamisen sekä hyödyntäminen osalta. Ohjelmistokehittäjiä ja testaajia osalta koulutusta on syytä täydentää ohjelmistometriikan välinekohtaisilla koulutuksilla. Koulutuksen järjestäminen edellyttää koulutusohjelman suunnittelun ja toteutuksen läpiviennin per kohderyhmä.

Kohdeyrityksessä tulisi järjestää pilotti ohjelmistometriikan hyödyntämisen sekä täsmennetyn toimintamallin osalta ennen laajempaa käyttöönottoa tai koulutuksia.

Edellä mainituille tehtäville tulee laatia sekä hyväksyä budjetti. Erilliselle kehittämissankkeelle tulee nimetä yksiselitteinen omistaja.

6 JOHTOPÄÄTÖKSET JA YHTEENVETO

Tutkimuksessa käsiteltiin ohjelmistometriikan soveltamista asiakaskohtaisesti räätälöidyn tietojärjestelmätoimituksen johtamisen apuvälineenä kohdeyrityksessä. Tutkimus perustui kohdeyrityksen tarpeeseen kehittää keinoja ennalta ennakoida sekä estää asiakaskohtaisesti räätälöityjen tietojärjestelmätoimituksien laatupoikkeamia.

Tutkimuksen päämääränä oli luoda lähtökohta kohdeyrityksessä tehtävälle ohjelmistometriikan hyödyntämisen määrämuotoistamiseen liittyvälle kehityshankkeelle sekä lähentää ohjelmistoliiketoiminnan johtamista ja ohjelmistokehitystä. Tärkein tutkimuksen tavoite oli luoda näkemys ohjelmistometriikan merkityksestä, hyödynnettävyydestä sijoittumisesta kohdeyrityksen toimintaan. Lisätavoitteena oli luoda käsitys ohjelmistometriikan hyödyntämisen edellytyksistä johtamisen apuvälineenä sekä myös ohjelmistometriikan käytäntöjen jalkautuksen lähtökohdista sekä mahdollista esteistä.

Tutkimus tuotti lopputuloksena kohdeyritykselle suosituksen suunnitella ja toteuttaa ohjelmistometriikan hyödyntämiseen liittyä kehitysprojekti. Tutkimuksen mukaan ohjelmistometriikan merkitys ja hyöty saavutetaan mahdollisuudesta minimoida laatukustannustekijöitä sekä parantaa tuottavuutta. Tutkimuksen aikana syntyi myös uusi malli (kuva 19), joka esittää ohjelmistometriikan sijoittumisen osaksi kohdeyrityksen toimintaan. Tutkimus tuotti tuloksena myös näkökulmia ohjelmistometriikan soveltamisesta johtamisen apuvälineenä toiminnan eri tasoilla sekä tähän liittyviä huomioitavia asioita sekä näkökulmia. Sähköisen kyselytutkimuksen lopputulosten analysoinnin perusteella tutkimus tuotti myös havaintoja ohjelmistometriikan nykytilasta sekä käyttöönottoon liittyvistä tekijöistä kohdeyrityksessä.

Tutkimuksen tekijänä uskon, että lähtökohtana ollut motiivi tutkimuksen tekemiseksi kohdeyrityksen ohjelmistoliiketoiminnan kannalta on yhä olemassa ja että olen tutkinut kohdeyrityksen kannalta hyödyllistä asiakokonaisuutta sekä tuottanut myös hyödyllisiä lopputuloksia. Tutkimuksen tavoitteiden, päämäärien sekä tutkimuskysymyksen osalta tutkimus on tuottanut työille asetetut lopputulokset sekä vastannut myös tutkimuskysymyksiin.

Tutkimuksen tekemisestä opin uusia asioita ohjelmistometriikasta sisällöllisesti kuin siihen liittyvästä kokonaisuudesta. Tutkimuksen yhteydessä löysin myös tutkimuksen aihepiirin teorioiden, tutkimuksen viitekehyksen sekä kohdeyrityksen toimintajärjestelmän avulla uusia näkökulmia, joita kykenen hyödyntämään omassa varsinaisessa työssäni. Tutkimuksen läpiviennin osalta opin erityisesti prosessimielessä käytännön tasolla, miten tiettyjä asioita kannattaa tehdä sekä mitä välttää, koska en ollut aiemmin toteuttanut vastaanlaista tutkimusta.

Jatkotutkimusaiheeksi esitän ohjelmistometriikan tutkimista asiakaskohtaisen tietojärjestelmätoimitusta asiakkaan näkökulmasta seuraavien aihealueiden osalta asiakokonaisuuksina laadun sekä kustannusten osalta tietojärjestelmän hankinta, käyttöönotto, tuotantokäyttö sekä ylläpito. Aihepiirin tutkiminen tilaajan näkökulmasta olisi luonnollinen jatkotutkimusaihe, koska tässä tutkimuksessa tarkastelin aihepiiriä toimittajan sisäisestä näkökulmasta.

JULKAISTUT LÄHTEET

Alain, Abran. 2010. Software Metrics and Software Metrology. ISBN 978-0-470-59720-0. Wiley 2010.

Ala-Mutka, Jukka. 2008. Strategiamalli. ISBN 9-789521-413124. Talentum 2008.

Ambler, Scott., Sadalce, Pramod. 2006. Refactoring Databases, Evolutionary Database Design. ISBN 0-32-129353-3. Addison-Wesley 2008.

Blankenship, Ed. Woodward, Matrin. Holliday, Grant. Keller, Brian. 2011. Professional Team Foundation Server 2010. ISBN 978-0-470-94332-8. Wrox 2011.

Brown, William. Malveau Raphael. McCormick III, Hays. Mobbray, Thomas. 1998. Anti Patterns. Refactoring Software, Architectures and Projects in Crisis. ISBN 0-471-19713-0. John Wiley & Sons 1998.

Capers, Jones. Bonsignour, Oliver. 2011. The Economics of Software Quality. ISBN 0-13-258220-1. Addison-Wesley 2011.

Capers, Jones. 2008. Applied Software Measurement. ISBN 0-13-258220-1. McGraw-Hill 2008.

Kamppinen, Matti. Kuusi, Osmo. Söderlund, Sari. 2002. Tulevaisuuden tutkimus ja perusteet. ISBN 951-746-389-8. Suomalaisen kirjallisuuden Seuran Toimituksia 896. Tallprint 2002.

Martin, Robert. 2009. Clean Code. A Handbook of Agile Software Craftsmanship. ISBN 978-0-13-235088-4. Prentice Hall 2009.

McConnell, Steve. 2006. Software Estimation, Demystifying the Back Art. ISBN 0-7356-0535-1. Microsoft Press 2006.

Microsoft Press. 2003. Analyzyzing Requirements and Refining Microsoft .NET Solution Architectures. ISBN 0-7356-1894.1. Microsoft Press 2003.

Pressman, Roger. 2000. Software Engineering, Apractioner's Approach, European Adaptation. ISBN 0-07-709677-0. McGraw Hill 2000.

Resnick, Steve. Bjork, Aaron. Maza, Michael. 2011. Professional Scrum with Team Foundation Server 2010. ISBN 978-0-470-94333-5. Wrox 2011.

Sipilä, Jorma. 1998. Asiantuntija ja asiakas - myymmekö tunteja vai tulosta? EKONIMIA-sarja, ISBN 951-0-23037-5, WSOY, Porvoo 1998.

Vesa, Suvi. 2011. Työhyvinvointi, organisaation menestys ja niiden yhdistäminen tietotyössä. Yhteiskunta- kulttuuritiedeiden yksikkö, Tampereen yliopisto. ISBN 978-951-44-8687-6 <http://reppu.lamk.fi/mod/resource/view.php?id=358612>.

Alain, Abran. Alain, April. 2008. Software Maintenance Management: Evaluation and Continous Improvement. ISBN 0470147075. Wiley 2008.

JULKAISEMATTOMAT LÄHTEET

Kohdeyrityksen toimintajärjestelmä. 2013. Kohdeyrityksen järjestelmä sisältäen sähköisessä muodossa prosessikuvaukset ja menetelmäohjeet: laatujärjestelmä, myyntiprosessi, toimitusprosessi, projektiprosessi ja systeemityömenetelmät.

SÄHKÖISET LÄHTEET

Agile Alliance. 2001. Manifesto for Agile Software Development. Saatavissa: <http://agilemanifesto.org>. Viitattu 3.5.2014.

Chambers & Associates Pty Ltd. 2014. McCabe Cyclomatic Complexity. Saatavissa: http://www.chambers.com.au/glossary/mc_cabe_cyclomatic_complexity.php. Viitattu 7.5.2014.

Denis, Meredith 2012. Best Practices in Software Metrics. Saatavissa: http://softwaresystemsbestpractices.com/Denis_Meredith_Best_Practices_in_Software_Metrics.php. Viitattu 6.10.2012.

FiSMA ry. 2014. Saatavissa: <http://www.fisma.fi>. Viitattu 3.5.2014.

Gack, Cary 2012. Software Metrics: The state of the Art. ITMPI maksullinen webinaari. Saatavissa: <http://www.itmpi.org/VideoDetail/tabid/736/VideoId/330/Default.aspx>. Viitattu 6.9.2012.

ISBSG. 2012. Saatavissa: <http://www.isbsg.org>. Viitattu 14.5.2012.

ITMPI. 2012. Saatavissa: <http://www.itmpi.org>. Viitattu 4.5.2012.

Jyväskylän Yliopisto. 2011. Menetelmäpolkuja humanisteille. Saatavissa: <https://koppa.jyu.fi/avoimet/hum/menetelmapolkuja/en/research-process/doingresearch#>. Viitattu 1.2.2012.

Kua, Patric. 2013. An Appropriate use of Metrics. Saatavissa: <http://martinfowler.com/articles/useOfMetrics.html>. Viitattu 2.5.2014.

Microsoft. 2014. Visual Studio2013 Code Metrics Output View. Saatavissa: <http://blogs.msdn.com>. Viitattu 18.4.2014.

MSDN. 2014. Saatavissa: <http://msdn.microsoft.com/en-us/library/bb385914.aspx>. Viitattu 29.4.2014.

Pietarinen, Ilmari. Sähköinen sanasto. Tietotekniikan liitto ry. Saatavissa: http://www.ttlry.fi/viikon_sana/katselmus-%E2%80%93review. Viitattu 20.1.2012.

QESP. 2014. Software Metrics. Saatavissa: <http://qesp.org/software-metrics>. Viitattu 11.4.2014.

Sharma, V.S., Kaulguld, V. 2013. Adoption and use of new metrics in a large organization: A case study. Saatavissa: <http://ieeexplore.ieee.org>. Viitattu 2.5.2014.

Smacchia, Patrick. 2014. NDepend 5 Dashboard. Saatavissa: <http://www.ndepend.com/Screenshots.aspx>. Viitattu 18.4.2014.

Technology & Business Integrators. 2012. Software Process Improvement and Metrics Project Abstract. Success Stories. Saatavissa: <http://www.tbicentral.com/success-stories/software-process-improvement-and-metrics-project-abstract>. Viitattu 20.5.2012

Westfall, Linda 2003. Are We Doing Well, Or Are We Doing Poorly. Artikkel. Saatavissa: http://www.westfallteam.com/Papers/Are_We_Doing_Well.pdf. Viitattu 14.10.2012.

Westfall, Linda 2005. 12 Steps to Useful Software Metrics. Saatavissa: http://www.westfallteam.com/Papers/12_steps_paper.pdf. Viitattu 15.10.2012.

Wikipedia. 2012. Software Engineering. Saatavissa: http://en.wikipedia.org/wiki/Software_engineering. Viitattu 6.3.2014.

LIITTEET

- 1) Sähköpostikutsu kyselytutkimukseen
- 2) Kyselytutkimuksen aloitusnäyttö
- 3) Kyselylomake
- 4) Kyselytutkimuksen vastausaineisto
- 5) Yhteenveto kyselytutkimuksen vastauksista
- 6) Kyselytutkimuksen kysymyskohtaiset graafit

Yksikön sisäinen tiedote,

OHJELMISTOTOIMITUKSEN JOHTAMISEEN UUSI APUVÄLINE

Kehitämme yksikössämme ohjelmistotuotantoon liittyviä käytäntöjä. Yksi kehityskohteemme on ohjelmistometriikan hyödyntäminen osana toimituksien sekä projektien johtamista.

Päämäärämme on lisätä ohjelmistotoimituksien teknisen laatutason läpinäkyvyyttä mittareiden avulla päätöksenteon tueksi, jotta voimme parantaa ohjelmistotuotantomme teknistä laatua sekä vähentää teknisen laadun poikkeamista johtuvia laatukustannustekijöitä tunnistamalla ongelmat riittävän ajoissa. Päämäärämme saavuttaminen vaatii metriikan osalta yhteisen toimintamallin luomisen sekä eri rooleissa toimivien henkilöiden osalta käytäntöjen, menetelmien sekä arkipäivän keinojen kehittämistä.

Käynnistän JOMA-kehittämistehtävän (johda ohjelmistometriikka apuvälineenä) alustavan selvityksen. Lopputuloksena tuotan ehdotuksen kehitysprojektista, minkä jälkeen yksikköemme johtoryhmä tekee päätöksen kehitysprojektin käynnistämisestä. Kehittämistehtävän aikana selvitän sähköisen kyselyn avulla ohjelmistometriikan hyödyntämisen nykytilan. Tämän lisäksi tunnistan kehittämistarpeita ohjelmistometriikkaan liittyvän teorian sekä nykyisten käytäntöjemme kautta.

Lähetän erikseen sähköpostitse linkin sähköiseen kyselyyn, mihin toivoin sinun vastaavan.

Ystävällisesti,

Arto Ihantoja

Home > Software Metrics

Software Metrics

All Sites

Home

View All Site Content

Surveys

SoftwareMetricsSurvey2013

Documents

Lists

Discussions

Sites

People and Groups

Recycle Bin

A site to share ideas of using software metrics

Ohjelmistometriikkakysely

Ohjelmistometriikkaan liittyvän kyselyni tarkoituksena on kartoittaa eri roolien näkökulmasta sisäisesti ohjelmistometriikan hyödyntämisen nykytilaa sekä siihen liittyviä asenteita tuotteesta, projektista tai palvelusta riippumatta. Kyselyn lopputulosta käytän toiminnankehittämiseen liittyvän kehitysehdotuksen laatimisessa.

Täytä ohjelmistometriikkakysely oman roolisi näkökulmasta nimettömästi 22.9.2013 mennessä.

Kyselyn voit täyttää useamman kerran per rooli, mikäli toimit useassa roolissa samanaikaisesti.

Klikkaa näytön vasemmasta reunasta **SoftwareMetricsSurvey2013** tai suoraan linkistä [http://\[URL\]/Lists/SoftwareMetricsSurvey2013/overview.aspx](http://[URL]/Lists/SoftwareMetricsSurvey2013/overview.aspx)

Seuravasta näytöstä aloitat kyselyn täyttämisen painikkeella **"Respond to this Survey"**.

Lisätietoja **Arto Ihantoja**.

101. Vastaa kyselyyn roolissa *

- ☒ 1. Arkkitehti
- ☐ 2. Määrittelijä
- ☐ 3. Suunnittelija
- ☐ 4. Toteuttaja
- ☐ 5. Testaaja
- ☐ 6. Projektipäällikkö
- ☐ 7. Palvelupäällikkö
- ☐ 8. Myyjä
- ☐ 9. Esimies
- ☐ 10. Johtaja
- ☐ 11. Muu

211. Tunnen ohjelmistometriikkaan liittyvää teoriaa ja käytäntöjä *

- ☐ 1. En tunne
- ☐ 2. Olen kuullut asiasta
- ☐ 3. Olen perehtynyt asiaan
- ☐ 4. Tunnen asian jossain määrin käytännön tasolla
- ☐ 5. Tunnen asian hyvin käytännön tasolla
- ☐ 6. Erikoisosaamiseni
- ☒ 0. Asia ei liity minuun

212. Olen saanut ohjelmistometriikkaan liittyvään koulutusta *

- ☐ 1. En ole saanut koulutusta
- ☐ 2. Olen perehtynyt itse
- ☐ 3. Olen käynyt peruskurssin
- ☐ 4. Olen opiskellut asiaa
- ☐ 5. Olen sertifioitu
- ☐ 6. Minulla on tutkinto
- ☒ 0. Asia ei liity minuun

213. Hyödynnän työssäni ohjelmistometriikkaa *

<div><input type="radio"/> 1. En hyödynnä</div> <div><input type="radio"/> 2. Hyödynnän erittäin harvoin</div> <div><input type="radio"/> 3. Hyödynnän ajoittain riippuen tilanteesta</div> <div><input type="radio"/> 4. Hyödynnän usein, mutta epäformaalisti</div> <div><input type="radio"/> 5. Hyödynnän usein, mutta formaalisti</div> <div><input type="radio"/> 6. Hyödynnän säännöllisesti siten, että se vaikuttaa myös päätöksiin</div> <div><input checked="" type="radio"/> 0. Asia ei liity minuun</div>
<p>214. Minulla on käytössäni ohjelmistometriikkaan liittyviä työvälineitä *</p> <div><input type="radio"/> 1. Ei ole</div> <div><input type="radio"/> 2. Käytössäni on yksinkertainen manuaalinen työväline, jolla pystyn tulkitsemaan tietoa</div> <div><input type="radio"/> 3. Käytössäni on yksinkertainen manuaalinen työväline, jolla pystyn tuottamaan tietoa</div> <div><input type="radio"/> 4. Käytössäni on työväline, jolla pystyn tulkitsemaan tietoa</div> <div><input type="radio"/> 5. Käytössäni on työväline, jolla pystyn tuottamaan tietoa</div> <div><input type="radio"/> 6. Käytössäni on automaattinen ammattikäyttöön tarkoitettu työväline</div> <div><input checked="" type="radio"/> 0. Asia ei liity minuun</div>
<p>215. Tarvitsen ohjelmistometriikkaan liittyvää koulutusta tai opastusta *</p> <div><input type="radio"/> 1. En tarvitse</div> <div><input type="radio"/> 2. Tarvitsen yleisesittelyn</div> <div><input type="radio"/> 3. Tarvitsen peruskoulutus tietojen tulkintaa varten</div> <div><input type="radio"/> 4. Tarvitsen peruskoulutuksen tietojen tuottamiseksi</div> <div><input type="radio"/> 5. Tarvitsen syvällisen koulutuksen tietojen tulkintaa varten</div> <div><input type="radio"/> 6. Tarvitsen syvällisen koulutuksen tietojen tuottamiseksi</div> <div><input checked="" type="radio"/> 0. Asia ei liity minuun</div>
<p>216. Esimieheni tai työnohjauksesta vastaava projektipäällikkö tai palvelupäällikkö tarvitsee ohjelmistometriikkaan tai sen hyödyntämiseen liittyvää koulutusta tai opastusta *</p> <div><input type="radio"/> 1. Ei tarvitse</div> <div><input type="radio"/> 2. Tarvitsee yleisesittelyn</div> <div><input type="radio"/> 3. Tarvitsee peruskoulutus tietojen tulkintaa varten</div> <div><input type="radio"/> 4. Tarvitsee peruskoulutuksen tietojen tuottamiseksi</div> <div><input type="radio"/> 5. Tarvitsee syvällisen koulutuksen tietojen tulkintaa varten</div> <div><input type="radio"/> 6. Tarvitsee syvällisen koulutuksen tietojen tuottamiseksi</div>

<input checked="" type="radio"/> 0. Asia ei liity esimieheeni
217. Alaiseni, työnohjauksessa olevat henkilöt tai työkaverini tarvitsevat ohjelmistometriikkaan liittyvää koulutusta tai opastusta * <input type="radio"/> 1. Eivät tarvitse <input type="radio"/> 2. Tarvitsevat yleisesittelyn <input type="radio"/> 3. Tarvitsevat peruskoulutuksen tietojen tulkintaa varten <input type="radio"/> 4. Tarvitsevat peruskoulutuksen tietojen tuottamiseksi <input type="radio"/> 5. Tarvitsevat syvällisen koulutuksen tietojen tulkintaa varten <input type="radio"/> 6. Tarvitsevat syvällisen koulutuksen tietojen tuottamiseksi <input checked="" type="radio"/> 0. Asia ei liity työympäristööni
218. Minulle tulee hankkia tai päivittää uusi ohjelmistometriikkaan liittyvä työväline * <input type="radio"/> 1. Ei tarvitse <input type="radio"/> 2. Tarvitsen manuaalisen perustyövälineen tietojen tulkintaan/seurantaan/päätöksen tekoon <input type="radio"/> 3. Tarvitsen manuaalisen perustyövälineen tietojen tuottamiseksi <input type="radio"/> 4. Tarvitsen automaattisen työvälineen tietojen tulkintaan/seurantaan/päätöksen tekoon <input type="radio"/> 5. Tarvitsen automaattisen työvälineen tietojen tuottamiseen <input type="radio"/> 6. Tarvitsen automaattisen ammattikäyttöön tarkoitetun työvälineen <input checked="" type="radio"/> 0. Asia ei liity minuun
219. Ohjelmistometriikan hyödyntäminen auttaisi minua saavuttamaan työssäni parempia tuloksia * <input type="radio"/> 1. Ei auta <input type="radio"/> 2. Auttaisi hyvin vähän (ei merkitystä) <input type="radio"/> 3. Auttaisi jossain määrin <input type="radio"/> 4. Auttaisi paljon <input type="radio"/> 5. Parantaisi nykyistä tilannetta merkittävästi <input type="radio"/> 6. Kriittinen tekijän asetun laatutavoitteen sekä asiakaslupauksen osalta <input checked="" type="radio"/> 0. Asia ei liity minuun
311. Olen nähnyt tarjouspyynnössä tai sen liitteissä asiakkaan vaatimuksia tai linjauksia ohjelmistometriikkaan liittyen * <input type="radio"/> 1. En ole nähnyt tai kuullut <input type="radio"/> 2. Olen nähnyt toiveena

- ☐ 3. Olen nähnyt työohjeen tai toimintamallin muodossa
- ☐ 4. Olen nähnyt osana laadun ohjausta
- ☐ 5. Olen nähnyt vaatimuksena
- ☐ 6. Olen nähnyt ehdottomana vaatimuksena
- ☒ 0. Asia ei liity minuun

312. Mielestäni ohjelmistometriikkaa tulee hyödyntää osana tarjousvalmistelu *

- ☐ 1. Ei tarvetta
- ☐ 2. Saattaisi olla hyötyä, mutten tiedä mitä
- ☐ 3. Olisi hyötyä ja minulla olisi myös ajatuksia hyödyntämistavasta
- ☐ 4. Tulisi hyödyntää vapaamuotoisesti tarjouksesta riippuen osana ratkaisun mitoittamista, työmääräarvioita sekä hinnoittelua
- ☐ 5. Tulisi hyödyntää systemaattisesti tarjouksesta riippuen osana ratkaisun mitoittamista, työmääräarvioita sekä hinnoittelua
- ☐ 6. Tulisi ehdottomasti aina hyödyntää sekä saada mukaan osaksi sopimusta
- ☒ 0. Asia ei liity minuun

313. Olen saanut ohjelmistometriikasta hyötyä tarjousvalmistelussa tai sopimus-neuvottelussa *

- ☐ 1. En ole hyötynyt
- ☐ 2. Saan hyvin harvoin ja erittäin vähän hyötyä
- ☐ 3. Saan harvoin hyötyä jonkin verran
- ☐ 4. Saan hyötyä usein
- ☐ 5. Saan merkittävää hyötyä satunnaisesti
- ☐ 6. Saan merkittävää hyötyä systemaattisesti
- ☒ 0. Asia ei liity minuun

314. Hyödynnän ohjelmistometriikkaa ratkaisun laajuuden, työmäärien sekä aika-taulujen arvioinnissa *

- ☐ 1. En hyödynnä
- ☐ 2. Hyödynnän erittäin harvoin
- ☐ 3. Hyödynnän harvoin epäformaalilla tavalla
- ☐ 4. Hyödynnän epäformaalilla tavalla
- ☐ 5. Hyödynnän formaalilla tavalla
- ☐ 6. Hyödynnän lähes aina systemaattisella tavalla
- ☒ 0. Asia ei liity minuun

315. Hyödynnän ohjelmistometriikkaa osana kaupallisen ehdotelman laatimista sekä hinnoittelua *

- ☐ 1. En hyödynnä
- ☐ 2. Hyödynnän erittäin harvoin
- ☐ 3. Hyödynnän harvoin epäformaalilla tavalla
- ☐ 4. Hyödynnän epäformaalilla tavalla
- ☐ 5. Hyödynnän formaalilla tavalla
- ☐ 6. Hyödynnän lähes aina systemaattisella tavalla
- ☒ 0. Asia ei liity minuun

411. Asetan etukäteen raja-arvot tehtävälle työlle tai perehdyn etukäteen annettuihin raja-arvoihin ennen työn tekemistä *

- ☐ 1. En aseta tai perehdy
- ☐ 2. Erittäin harvoin
- ☐ 3. Satunnaisesti
- ☐ 4. Usein
- ☐ 5. Lähes aina (tilannekohtaisesti harkiten)
- ☐ 6. Systemaattisesti aina
- ☒ 0. Asia ei liity minuun

412. Seuraan työn tekemisen aikana etukäteen asetettuja raja-arvoja *

- ☐ 1. En aseta tai perehdy
- ☐ 2. Erittäin harvoin
- ☐ 3. Satunnaisesti
- ☐ 4. Usein
- ☐ 5. Lähes aina (tilannekohtaisesti harkiten)
- ☐ 6. Systemaattisesti aina
- ☒ 0. Asia ei liity minuun

413. Korjaan työn tekemisen aikana raja-arvojen perusteella työn laatua tai suoritan muita korjaavia toimenpiteitä *

- ☐ 1. En korjaa
- ☐ 2. Erittäin harvoin
- ☐ 3. Satunnaisesti
- ☐ 4. Usein
- ☐ 5. Lähes aina (tilannekohtaisesti harkiten)

<p><input type="radio"/> 6. Systemaattisesti aina</p> <p><input checked="" type="radio"/> 0. Asia ei liity minuun</p>
<p>414. Todennan työn tekemisen jälkeen työlle asetettujen raja-arvojen toteutumista sekä ilmoitan tarvittaessa poikkeamasta *</p> <p><input type="radio"/> 1. En koskaan</p> <p><input type="radio"/> 2. Erittäin harvoin</p> <p><input type="radio"/> 3. Ajoittain (pääsääntöisesti en)</p> <p><input type="radio"/> 4. Usein</p> <p><input type="radio"/> 5. Lähes aina (tilannekohtaisesti harkiten)</p> <p><input type="radio"/> 6. Systemaattisesti aina</p> <p><input checked="" type="radio"/> 0. Asia ei liity minuun</p>
<p>415. Hyödynnän ohjelmistometriikkaa oman tai muiden työni ohjaamisessa ennen työtä, työn tekemisen aikana tai työn tekemisen jälkeen *</p> <p><input type="radio"/> 1. En koskaan</p> <p><input type="radio"/> 2. Erittäin harvoin</p> <p><input type="radio"/> 3. Ajoittain (pääsääntöisesti en)</p> <p><input type="radio"/> 4. Usein</p> <p><input type="radio"/> 5. Lähes aina (tilannekohtaisesti harkiten)</p> <p><input type="radio"/> 6. Systemaattisesti aina</p> <p><input checked="" type="radio"/> 0. Asia ei liity minuun</p>
<p>416. Seuraan sekä ohjaan ohjelmistometriikan avulla alihankkijoiden, near shore, offshoren tai blended shore työn etenemistä sekä tuloksia *</p> <p><input type="radio"/> 1. En seuraa tai ohjaa</p> <p><input type="radio"/> 2. Erittäin harvoin</p> <p><input type="radio"/> 3. Satunnaisesti</p> <p><input type="radio"/> 4. Usein</p> <p><input type="radio"/> 5. Lähes aina (tilannekohtaisesti harkiten)</p> <p><input type="radio"/> 6. Systemaattisesti aina</p> <p><input checked="" type="radio"/> 0. Asia ei liity minuun</p>
<p>417. Kerään ohjelmistometriikkatiedon myöhempää hyödyntämistä varten *</p> <p><input type="radio"/> 1. En kerää</p> <p><input type="radio"/> 2. Erittäin harvoin</p>

<p><input type="radio"/> 3. Satunnaisesti</p> <p><input type="radio"/> 4. Usein</p> <p><input type="radio"/> 5. Lähes aina (tilannekohtaisesti harkiten)</p> <p><input type="radio"/> 6. Systemaattisesti aina</p> <p><input checked="" type="radio"/> 0. Asia ei liity minuun</p>
<p>418. Pyrin oppimaan aiemmin kerätystä metriikkatiedosta tai hyödyntämään sitä muutoin *</p> <p><input type="radio"/> 1. En koskaan</p> <p><input type="radio"/> 2. Erittäin harvoin</p> <p><input type="radio"/> 3. Ajoittain (pääsääntöisesti en)</p> <p><input type="radio"/> 4. Usein</p> <p><input type="radio"/> 5. Lähes aina (tilannekohtaisesti harkiten)</p> <p><input type="radio"/> 6. Systemaattisesti aina</p> <p><input checked="" type="radio"/> 0. Asia ei liity minuun</p>
<p>419. Minulla käytettävissäni dokumentoitu malli ja tapa hyödyntää ohjelmistometriikkaa *</p> <p><input type="radio"/> 1. Ei ole</p> <p><input type="radio"/> 2. Projektikohtainen ohje</p> <p><input type="radio"/> 3. Tiimikohtainen ohje</p> <p><input type="radio"/> 4. Yksikkökohtainen ohje</p> <p><input type="radio"/> 5. Yhtiökohtainen ohje</p> <p><input checked="" type="radio"/> 6. Asia ei liity minuun</p>
<p>511. Haluan osallistua ohjelmistometriikkaan liittyvään koulutukseen *</p> <p><input type="radio"/> 1. En halua</p> <p><input type="radio"/> 2. Kyllä, vain yleistietoisku hyödyntämisen osalta</p> <p><input type="radio"/> 3. Kyllä, yleistietoisku tuottamisen osalta</p> <p><input type="radio"/> 4. Kyllä, tietoisku ja koulutukset hyödyntämisen ja tuottamisen osalta</p> <p><input type="radio"/> 5. Kyllä, tietoisku, koulutukset sekä käytännön työpajat hyödyntämisen osalta</p> <p><input type="radio"/> 6. Kyllä, tietoisku, koulutukset sekä käytännön työpajat tuottamisen osalta</p> <p><input checked="" type="radio"/> 0. Asia ei liity minuun</p>
<p>512. Haluan osallistua tuottamaan ohjelmistometriikkaan liittyvää koulutusta tai käytäntöjen kehittämistä. *</p>

- ☐ 1. En halua
- ☐ 2. Kyllä, vain yleistietoisku hyödyntämisen osalta
- ☐ 3. Kyllä, yleistietoisku tuottamisen osalta
- ☐ 4. Kyllä, tietoisku ja koulutukset tuottamisen ja hyödyntämisen osalta
- ☐ 5. Kyllä, tietoisku, koulutukset sekä käytännön työpajat hyödyntämisen osalta
- ☐ 6. Kyllä, tietoisku, koulutukset sekä käytännön työpajat tuottamisen osalta
- ☒ 0. Asia ei liity minuun

513. Saan minua johtavilta tukea sekä kannustusta kehittää työtäni parempien lopputulosten saavuttamiseksi *

- ☐ 1. En koskaan
- ☐ 2. Erittäin harvoin
- ☐ 3. Ajoittain (pääsääntöisesti en)
- ☐ 4. Usein
- ☐ 5. Lähes aina (tilannekohtaisesti harkiten)
- ☐ 6. Systemaattisesti aina
- ☒ 0. Asia ei liity minuun

514. Mielestäni ohjelmistometriikan hyödyntäminen johtamisen osana sen suurimpana esteenä tai haasteena on *

- ☐ 1. Ei estettä
- ☐ 2. Tietämättömyys/osaamattomuus
- ☐ 3. Tietojen keräämisen vaikeus
- ☐ 4. Käytettävissä oleva aika tai energia
- ☐ 5. Työvälineiden puuttuminen
- ☐ 6. Toimintamallit tai prosessit
- ☒ 0. Asia ei liity minuun

515. Suosittelen panostamaan ohjelmistometriikkaan hyödyntämiseen osana johtamista erityisesti *

- ☐ 1. En suosittelen
- ☐ 2. Suosittelen teknisille henkilöille
- ☐ 3. Suosittelen projektitiimille
- ☐ 4. Suosittelen projektijohdolle tai työnohjaukselle
- ☐ 5. Suosittelen esimiehille, johdolle ja myynnille



6. Suosittelen kaikille

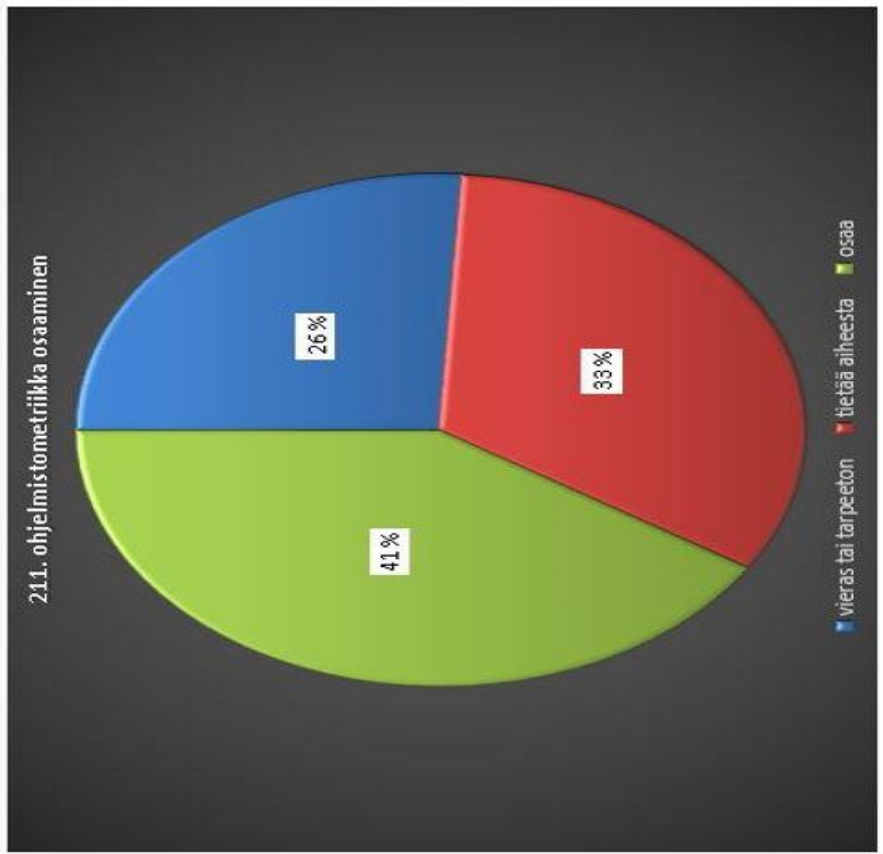
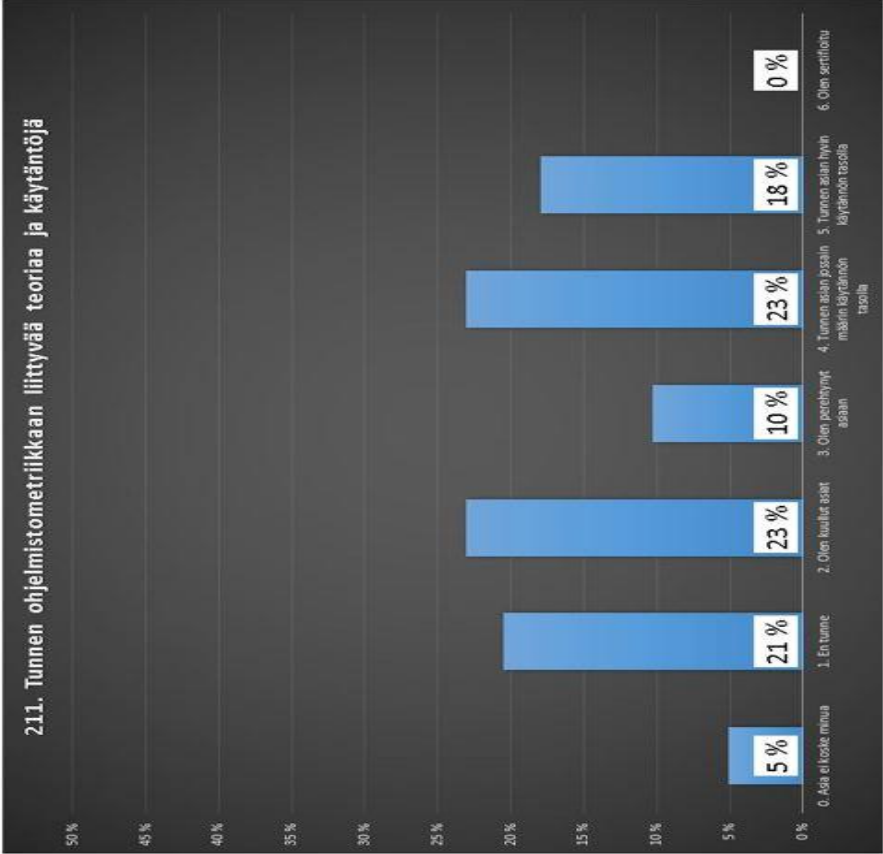


0. Asia ei liity minuun

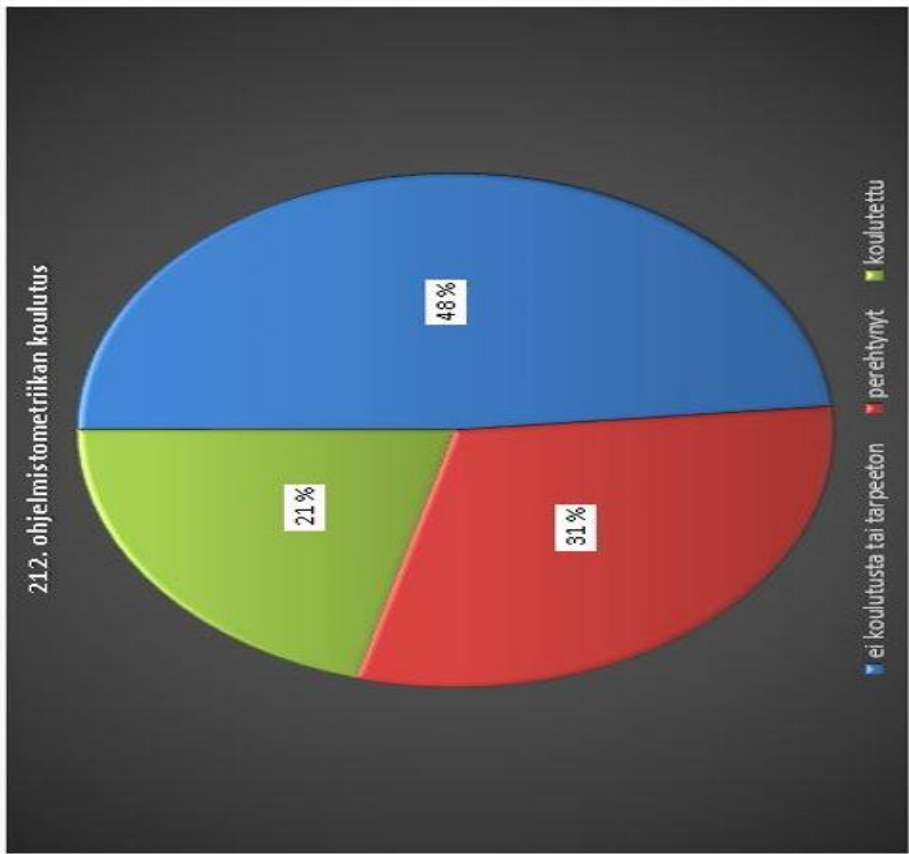
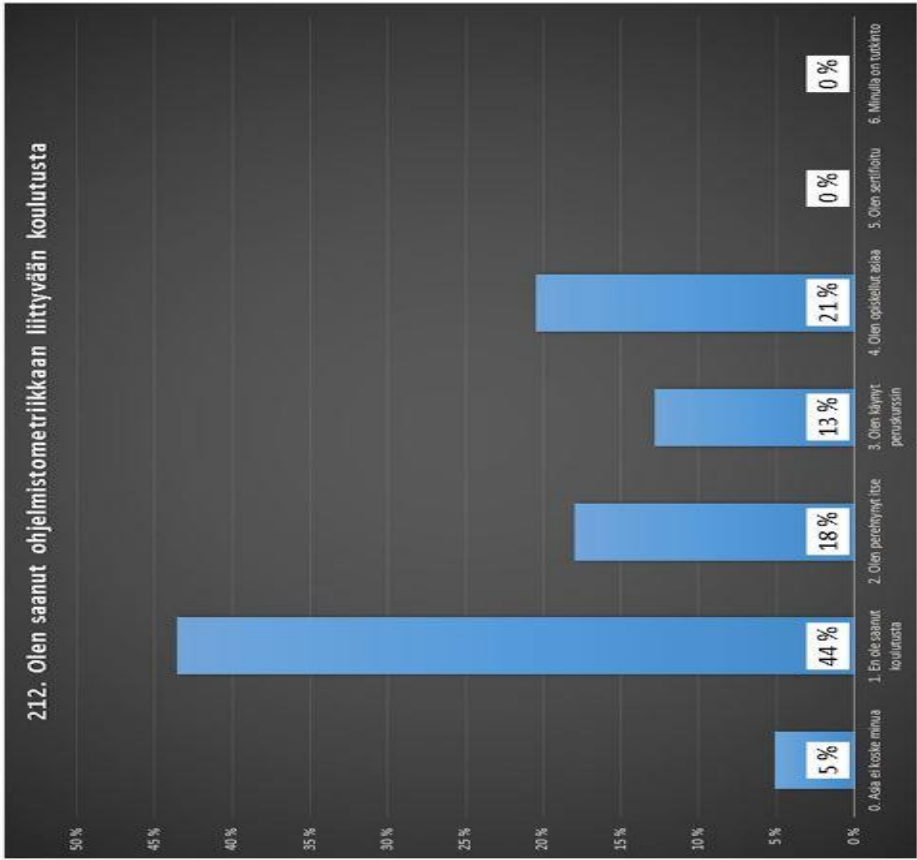
Date	101	211	212	213	214	215	216	217	218	219	311	312	313	314	315	411	412	413	414	415	416	417	418	419	511	512	513	514	515
9.9.2013	4	1	1	1	1	2	1	0	0	0	1	2	0	1	1	3	3	1	0	0	0	0	0	6	0	0	3	0	0
9.9.2013	4	4	1	3	5	1	1	0	1	2	1	4	1	1	1	1	1	1	1	2	1	1	1	1	1	1	2	2	3
9.9.2013	10	5	2	3	1	2	3	6	6	5	5	5	3	3	2	1	2	2	2	2	2	2	2	1	2	1	1	2	5
9.9.2013	4	0	4	3	3	6	0	5	0	6	1	0	0	3	0	1	2	3	1	3	0	2	3	1	5	5	2	6	4
9.9.2013	9	5	2	4	5	5	3	6	6	5	0	4	3	3	3	3	3	3	3	3	1	3	4	5	6	6	5	0	5
10.9.2013	10	2	1	1	1	3	2	4	1	4	1	2	1	1	1	1	1	0	1	1	1	1	1	1	3	0	3	2	4
12.9.2013	3	1	1	1	1	4	2	0	3	4	1	2	0	1	0	2	2	1	1	1	0	1	1	1	4	1	4	2	3
16.9.2013	1	3	3	3	2	5	0	2	4	3	4	4	3	4	4	3	4	3	4	4	0	3	4	1	3	2	5	6	3
16.9.2013	2	2	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	6	0	0	4	0	0	
16.9.2013	1	3	3	3	4	5	3	4	5	3	0	0	0	2	0	2	2	2	2	2	2	1	2	1	6	0	2	4	3
16.9.2013	6	4	2	3	1	1	0	6	1	3	4	2	1	2	0	5	6	1	5	1	1	1	1	2	1	3	3	6	4
16.9.2013	1	5	4	4	6	1	3	3	1	0	5	5	5	4	4	3	5	5	5	3	3	3	4	2	1	6	5	2	5
16.9.2013	9	4	4	2	1	2	2	3	1	3	3	3	1	1	1	0	0	0	0	0	0	0	0	1	3	1	4	2	3
16.9.2013	1	4	2	2	1	6	3	3	6	5	2	2	0	0	0	1	2	2	2	2	0	1	2	1	5	1	4	2	3
16.9.2013	4	4	4	4	5	5	5	5	6	6	0	0	0	3	3	3	3	3	3	2	0	2	2	1	5	0	5	2	3
16.9.2013	7	2	1	1	1	1	1	1	1	1	1	1	1	1	1	6	6	6	6	0	0	0	4	6	0	0	6	0	0
17.9.2013	4	5	4	4	6	1	3	3	6	6	0	0	0	4	0	5	5	5	4	4	0	4	4	2	5	5	5	6	6
17.9.2013	6	4	3	2	1	5	3	3	1	6	4	4	3	4	3	6	6	6	6	6	0	2	0	1	5	1	6	6	3
17.9.2013	5	1	1	1	1	1	1	1	1	1	0	0	0	0	0	3	3	3	0	2	3	0	1	0	1	1	4	2	2
17.9.2013	2	1	1	1	1	1	1	1	1	0	0	0	0	0	0	2	2	2	1	0	0	0	0	1	1	4	0	0	0
17.9.2013	6	1	1	1	1	2	0	2	0	0	1	0	0	1	1	5	5	3	5	0	0	3	6	1	2	0	3	2	0
17.9.2013	10	3	3	1	1	2	2	3	1	1	2	5	0	0	0	5	5	5	5	0	0	0	0	6	2	0	6	4	3
17.9.2013	7	2	1	1	1	3	3	4	2	4	2	5	0	0	1	3	3	2	2	2	3	2	1	1	5	1	1	2	3
17.9.2013	1	4	2	2	3	6	2	3	6	5	1	2	1	1	1	1	1	1	1	0	1	1	1	5	0	4	2	2	3
18.9.2013	5	1	1	1	1	2	2	2	1	1	0	0	1	0	0	3	3	3	2	0	0	0	1	1	2	1	5	2	2
18.9.2013	4	5	4	4	5	5	3	2	6	4	0	0	0	2	0	3	3	2	2	2	2	1	1	1	5	4	5	4	5
18.9.2013	6	2	1	1	1	1	1	1	1	1	1	1	1	1	1	6	6	6	6	1	1	1	5	1	1	5	1	2	1
18.9.2013	5	2	1	1	1	2	2	2	0	0	0	0	0	0	0	6	6	6	6	0	0	0	6	6	2	1	6	6	2
18.9.2013	6	4	2	2	1	3	2	3	1	4	3	3	3	3	3	6	6	6	6	6	1	1	1	5	1	5	1	6	2
19.9.2013	3	1	1	1	1	2	2	2	1	2	0	0	0	0	1	0	3	3	3	1	1	1	2	1	2	1	5	1	2
19.9.2013	5	1	0	0	1	0	1	1	1	0	0	0	0	0	0	4	4	4	5	0	0	4	4	1	0	0	5	0	2
19.9.2013	5	2	1	2	1	3	3	0	1	2	1	1	1	1	1	4	4	4	4	0	1	2	3	1	2	1	6	2	2
19.9.2013	2	0	0	0	0	2	2	2	1	0	0	0	0	0	0	2	2	2	2	0	0	0	0	1	1	4	1	1	1
19.9.2013	2	2	1	0	1	2	2	2	1	2	0	0	0	0	0	2	2	2	2	2	1	0	0	6	2	1	4	2	2
20.9.2013	3	2	1	1	1	2	2	2	1	0	0	2	0	0	0	2	2	2	2	0	0	0	2	1	4	1	4	2	2
20.9.2013	4	3	2	2	3	5	2	2	6	5	2	0	0	3	0	2	2	2	2	2	0	2	3	1	5	0	5	2	2
20.9.2013	4	5	4	4	6	1	5	1	6	6	3	4	2	4	2	4	4	4	4	3	0	4	4	2	5	6	5	2	4
20.9.2013	1	5	4	4	6	1	3	3	6	6	1	3	3	4	4	4	4	4	4	3	1	3	3	2	5	3	5	2	6
21.9.2013	4	4	3	4	4	5	3	3	6	0	0	3	1	4	0	3	3	3	3	2	1	3	4	1	6	3	5	2	5

YHTEENVETO KYSELYTUTKIMUKSEN VASTAUKSISTA

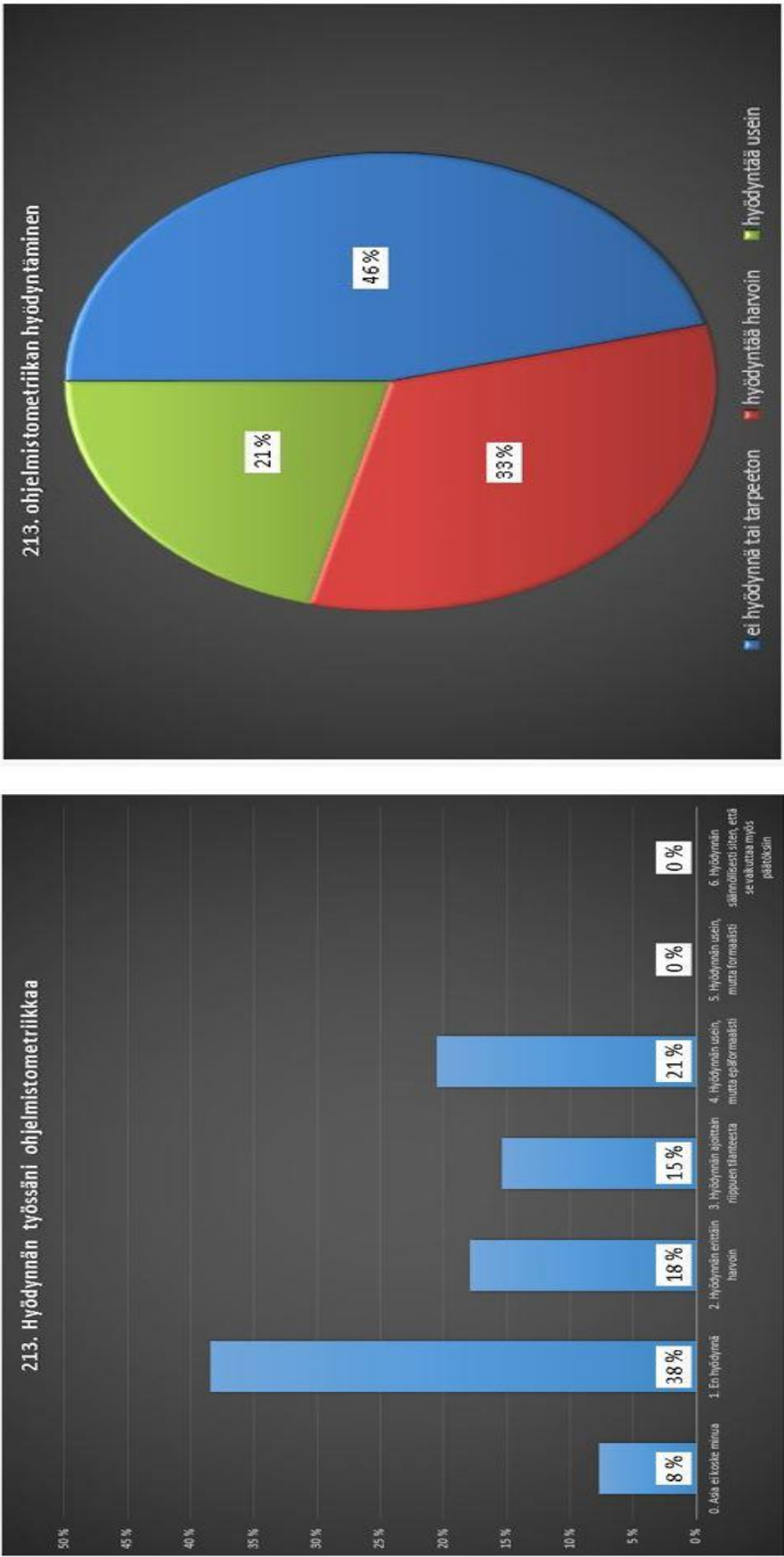
[illegible]



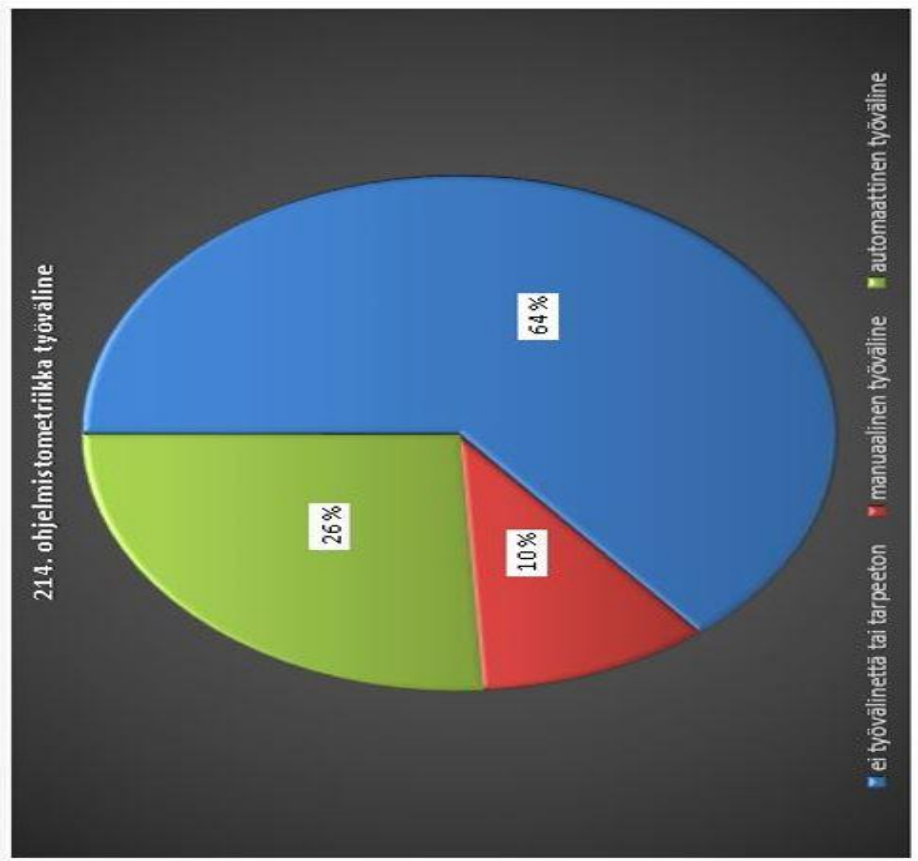
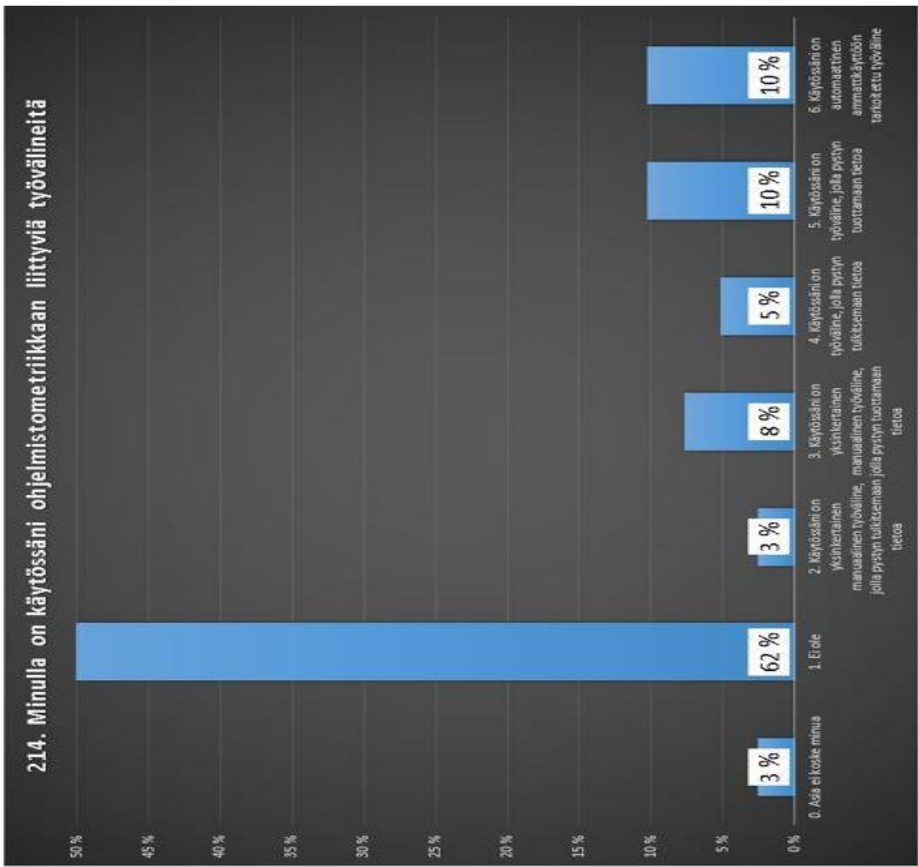
0. Asia ei koske minua	5 %	2	26 %	vieras tai tarpeeton (0-1)
1. En tunne	21 %	8		
2. Olen kuullut asiat	23 %	9	33 %	tietää aiheesta (2-3)
3. Olen perehtynyt asiaan	10 %	4		
4. Tunnen asian jossain määrin käytännön tasolla	23 %	9	41 %	osaa (4-6)
5. Tunnen asian hyvin käytännön tasolla	18 %	7		
6. Olen sertifioitu	0 %	0		



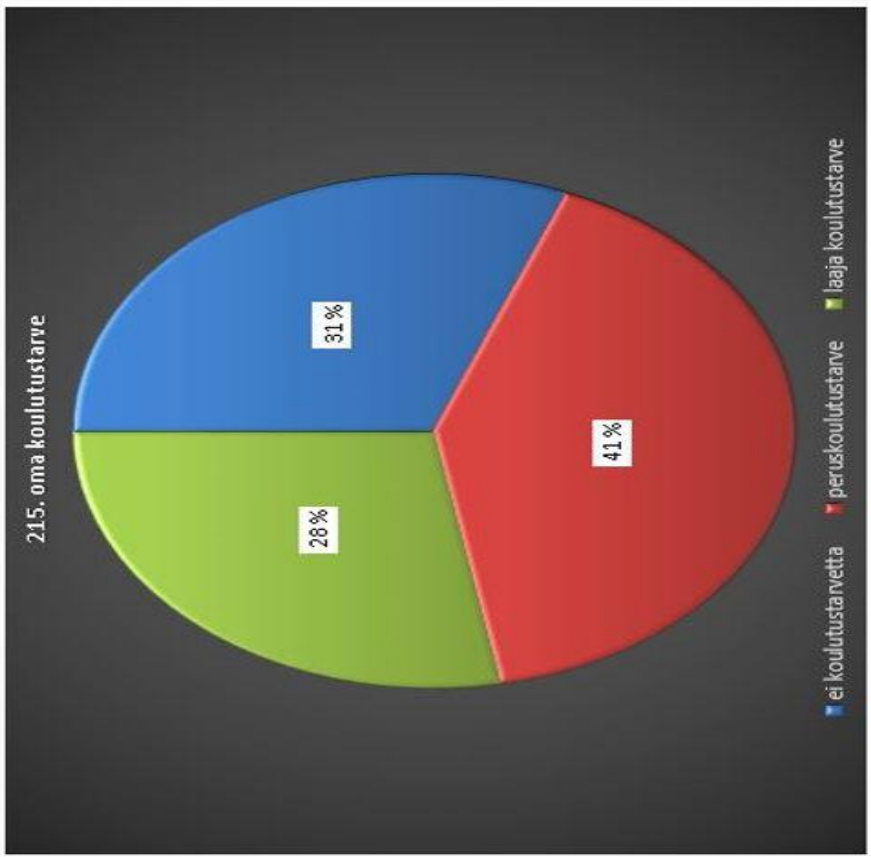
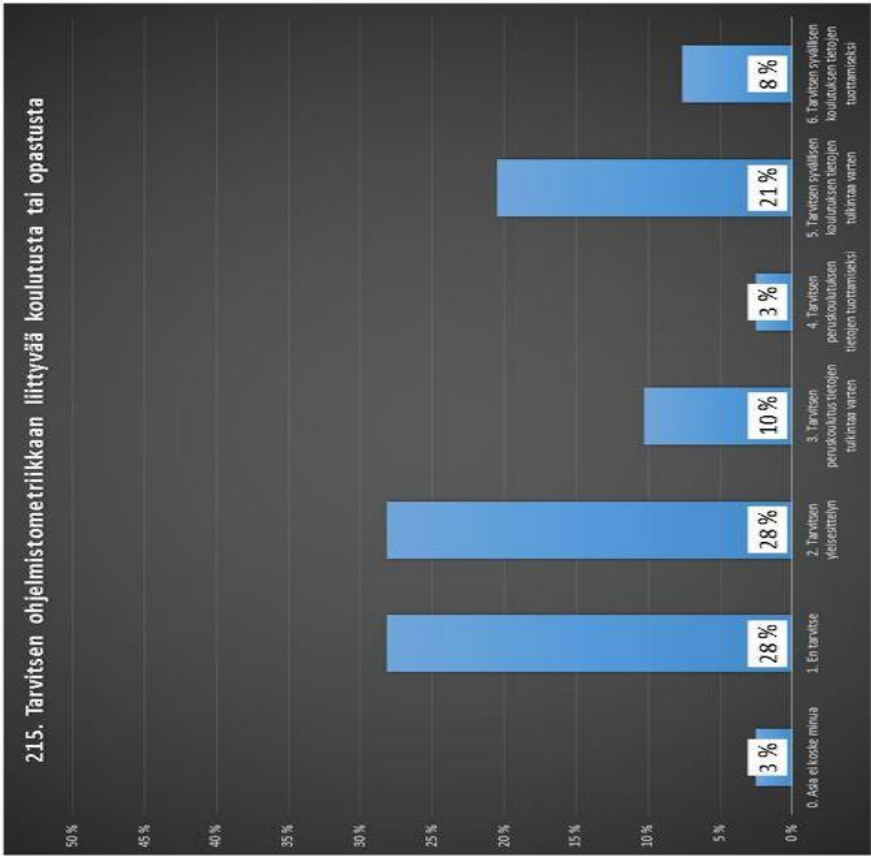
0. Asia ei koske minua	5 %	2	49 %	ei koulutusta tai tarpeeton (0-1)
1. En ole saanut koulutusta	44 %	17		
2. Olen perehtynyt itse	18 %	7	31 %	perehtynyt (2-3)
3. Olen käynyt peruskurssin	13 %	5		
4. Olen opiskellut asiaa	21 %	8	21 %	koulutettu (4-6)
5. Olen sertifioitu	0 %	0		
6. Minulla on tutkinto	0 %	0		



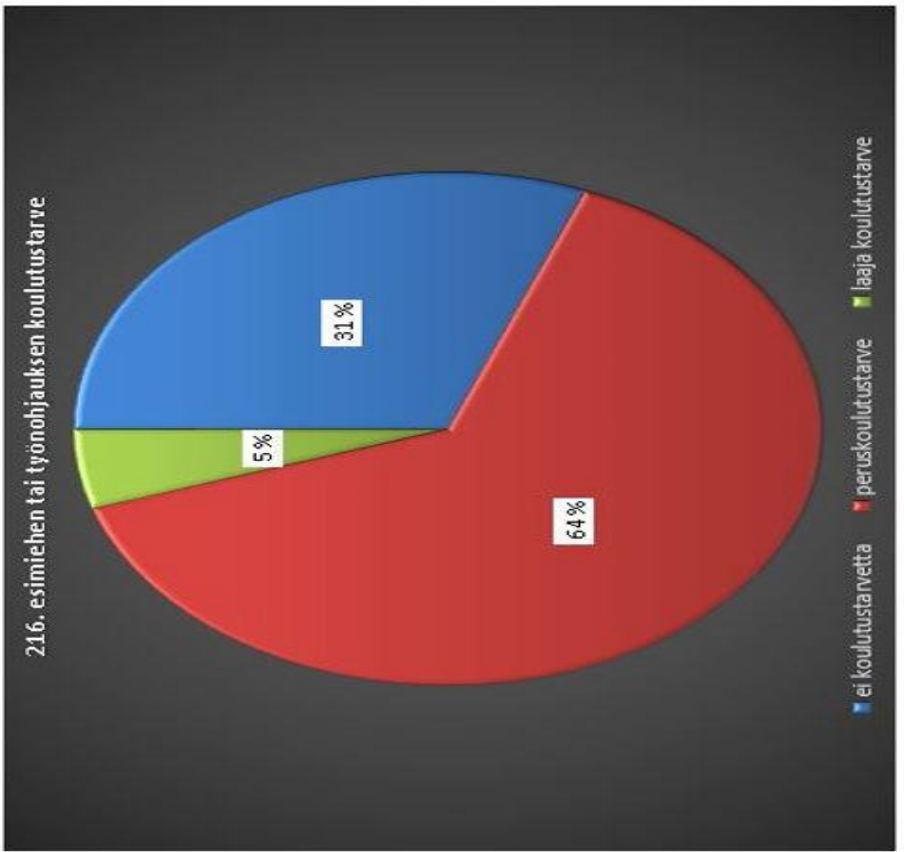
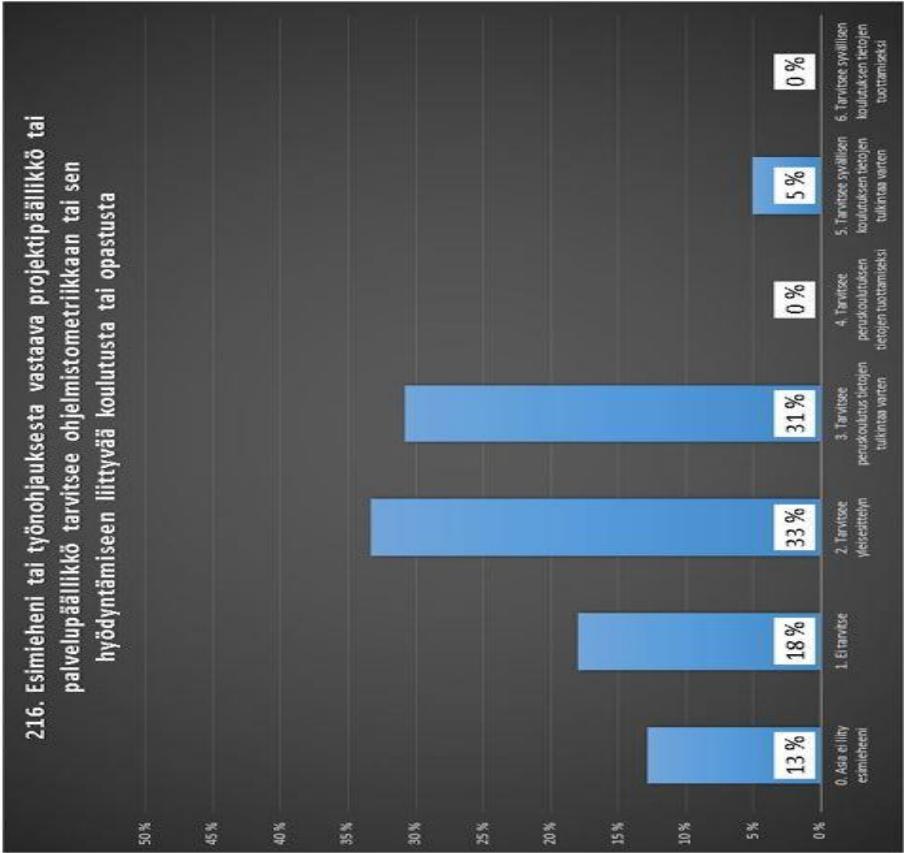
0. Asia ei koske minua	8 %	3	46 % ei hyödynnä tai tarpeeton (0-1)
1. En hyödynnä	38 %	15	
2. Hyödynnän erittäin harvoin	18 %	7	33 % hyödyntää harvoin (2-3)
3. Hyödynnän ajoittain riippuen tilanteesta	15 %	6	
4. Hyödynnän usein, mutta epäformaalisti	21 %	8	21 % hyödyntää(4-6)
5. Hyödynnän usein, mutta formaalisti	0 %	0	
6. Hyödynnän säännöllisesti siten, että se vaikuttaa myös päätöksiin	0 %	0	



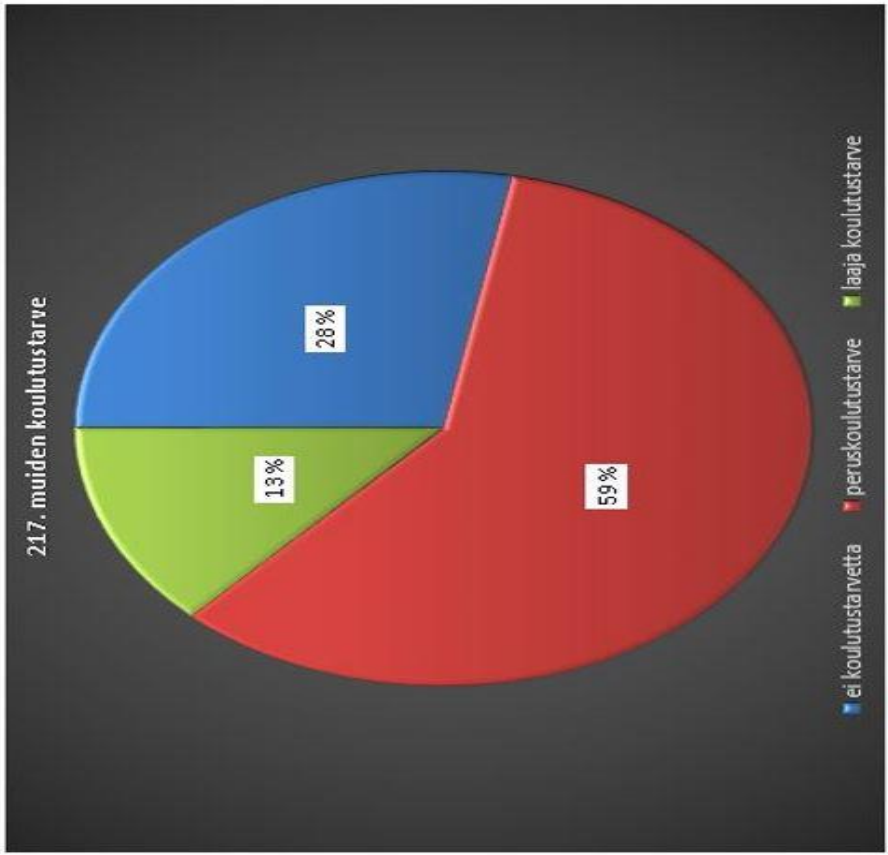
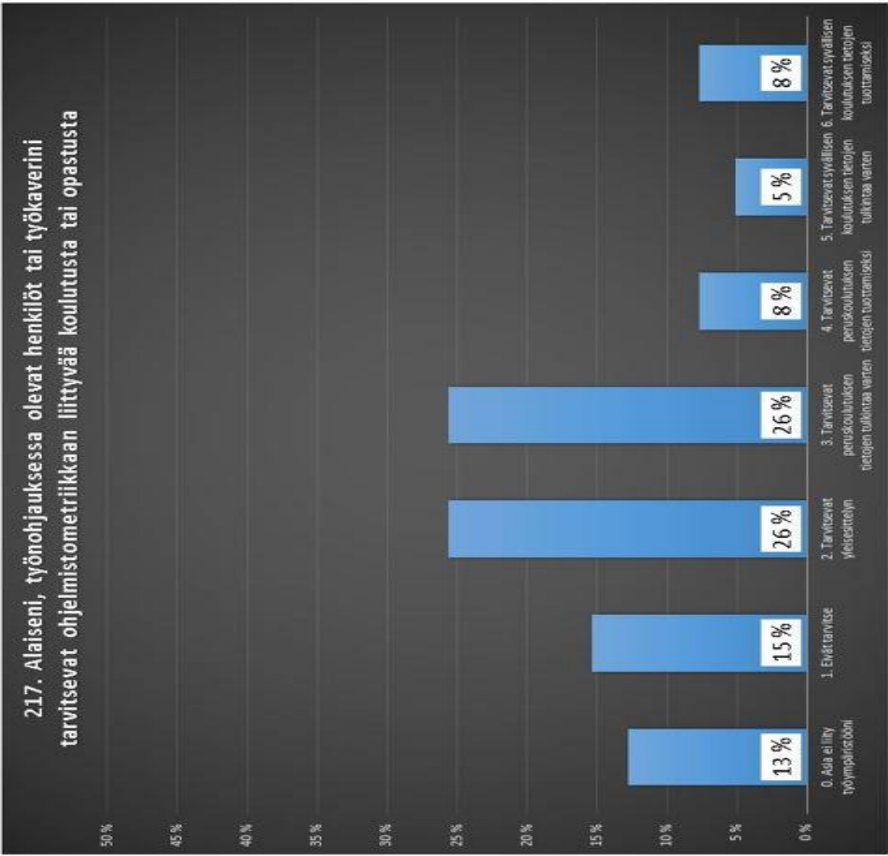
0. Asia ei koske minua	3 %	1	64 % ei työvälinettä tai tarpeeton (0-1)
1. Ei ole	62 %	24	
2. Käytössäni on yksinkertainen manuaalinen työväline, jolla pystyn tulkitsemaan tietoa	3 %	1	10 % manuaalinen työväline (2-3)
3. Käytössäni on yksinkertainen manuaalinen työväline, jolla pystyn tuottamaan tietoa	8 %	3	
4. Käytössäni on työväline, jolla pystyn tulkitsemaan tietoa	5 %	2	26 % automaattinen työväline (4-6)
5. Käytössäni on työväline, jolla pystyn tuottamaan tietoa	10 %	4	
6. Käytössäni on automaattinen ammattikäyttöön tarkoitettu työväline	10 %	4	



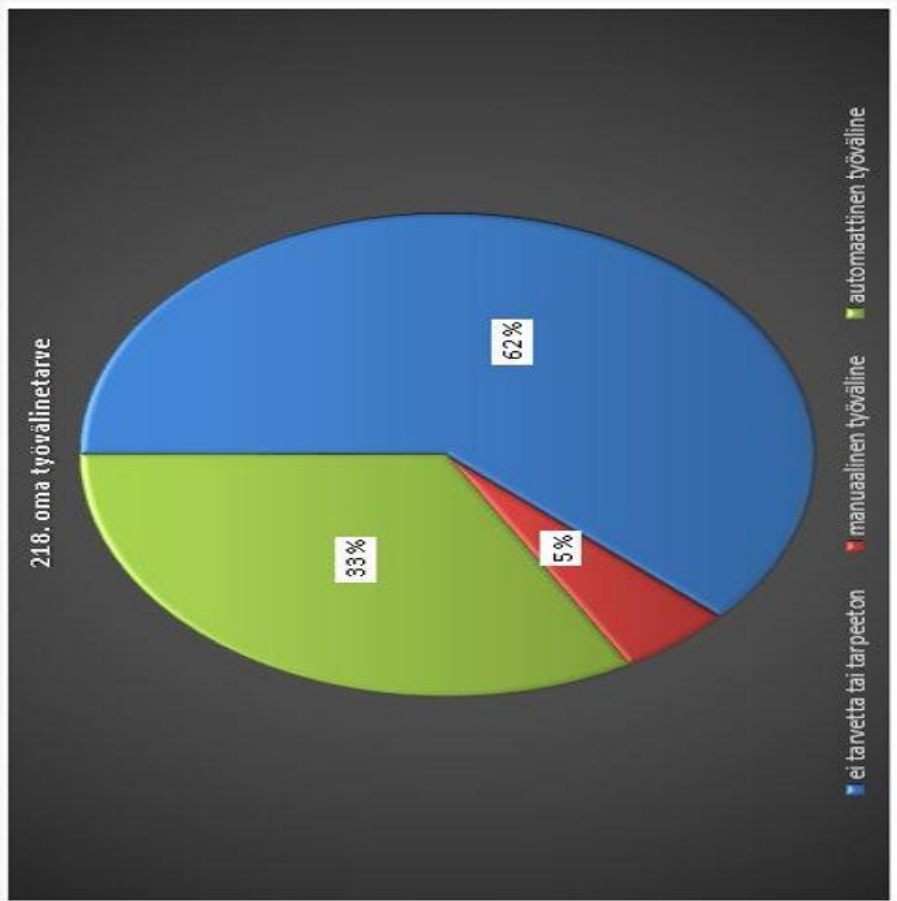
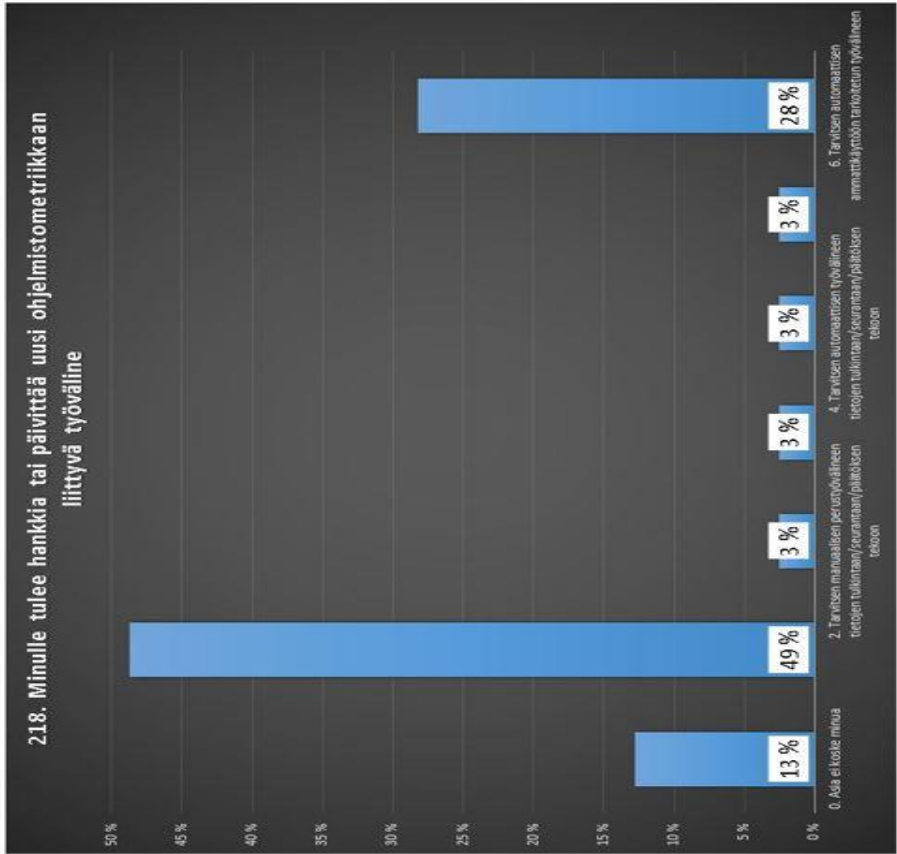
0. Asia ei koske minua	3 %	1	31 %	ei tarvitse koulutusta (0-1)
1. En tarvitse	28 %	11		
2. Tarvitsen yleisesittelyyn	28 %	11	41 %	peruskoulutustarve (2-4)
3. Tarvitsen peruskoulutuksen tietojen tulkintaa varten	10 %	4		
4. Tarvitsen peruskoulutuksen tietojen tuottamiseksi	3 %	1		
5. Tarvitsen syvällisen koulutuksen tietojen tulkintaa varten	21 %	8	28 %	laaja koulutustarve (5-6)
6. Tarvitsen syvällisen koulutuksen tietojen tuottamiseksi	8 %	3		



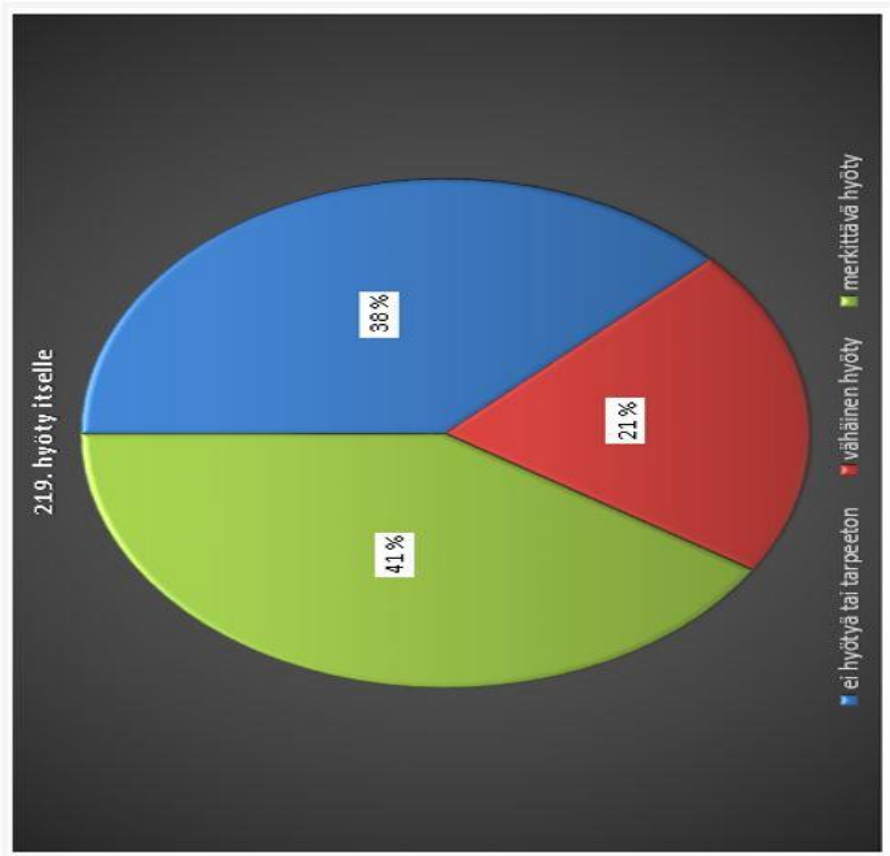
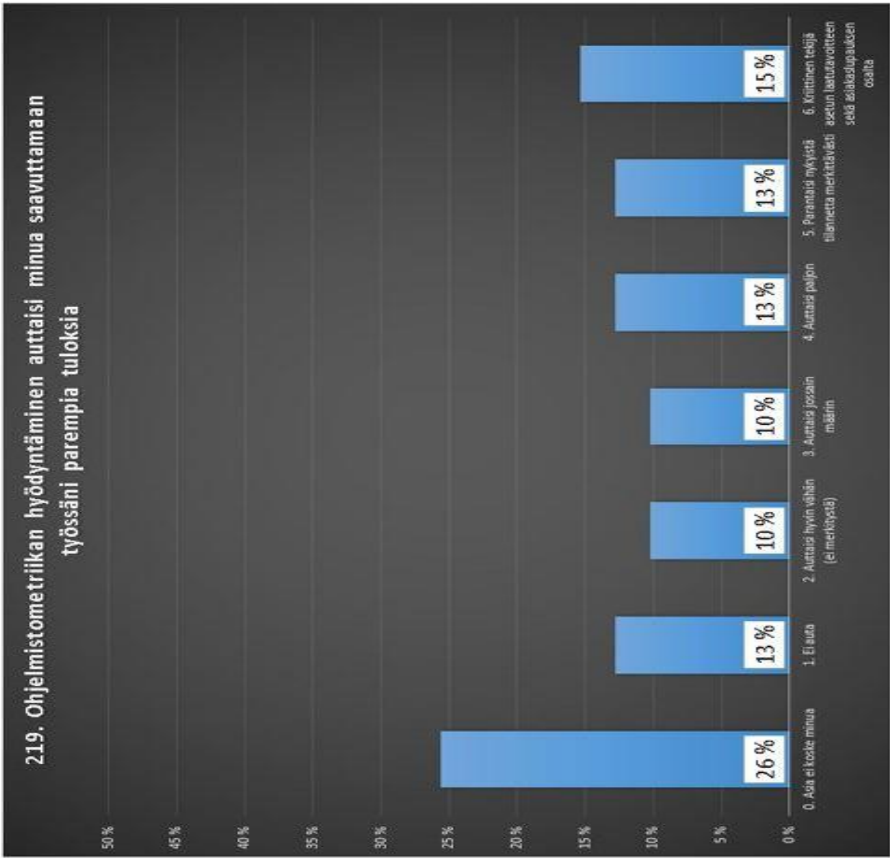
0. Asia ei liity esimieheeni	13 %	5	31 %	ei tarvetta tai tarpeeton (0-1)
1. Ei tarvitse	18 %	7		
2. Tarvitsee yleisestitlyn	33 %	13	64 %	peruskoulutustarve (2-4)
3. Tarvitsee peruskoulutuksen tietojen tulkintaa varten	31 %	12		
4. Tarvitsee peruskoulutuksen tietojen tuottamiseksi	0 %	0		
5. Tarvitsee syvällisen koulutuksen tietojen tulkintaa varten	5 %	2	5 %	laaja koulutustarve (5-6)
6. Tarvitsee syvällisen koulutuksen tietojen tuottamiseksi	0 %	0		



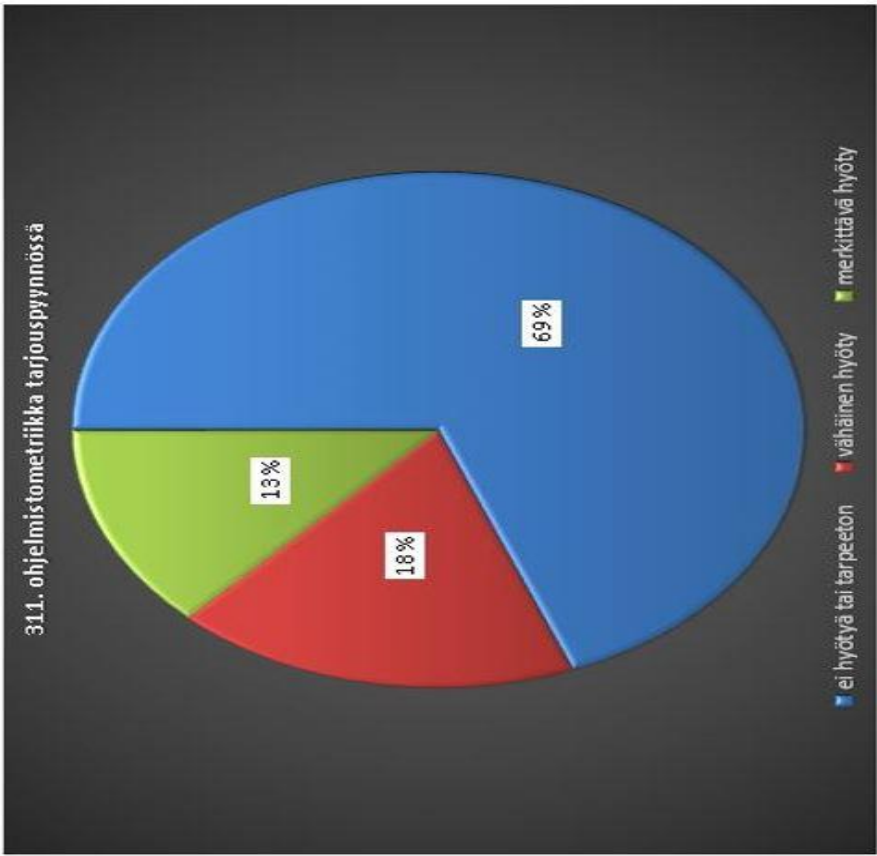
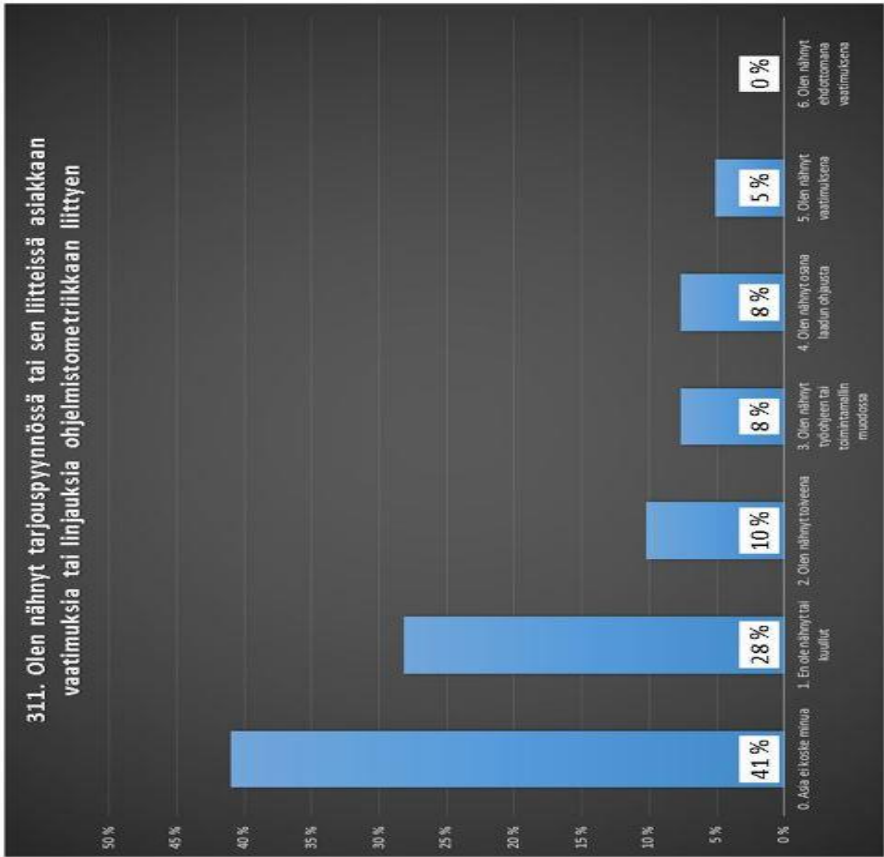
0. Asia ei liity työympäristöni	13 %	5	28 %	ei tarvetta tai tarpeeton (0-1)
1. Eivät tarvitse	15 %	6		
2. Tarvitsevat yleisesittelyn	26 %	10	59 %	peruskoulutustarve (2-4)
3. Tarvitsevat peruskoulutuksen tietojen tulkintaa varten	26 %	10		
4. Tarvitsevat peruskoulutuksen tietojen tuottamiseksi	8 %	3		
5. Tarvitsevat syvällisen koulutuksen tietojen tulkintaa varten	5 %	2	13 %	laaja koulutustarve (5-6)
6. Tarvitsevat syvällisen koulutuksen tietojen tuottamiseksi	8 %	3		



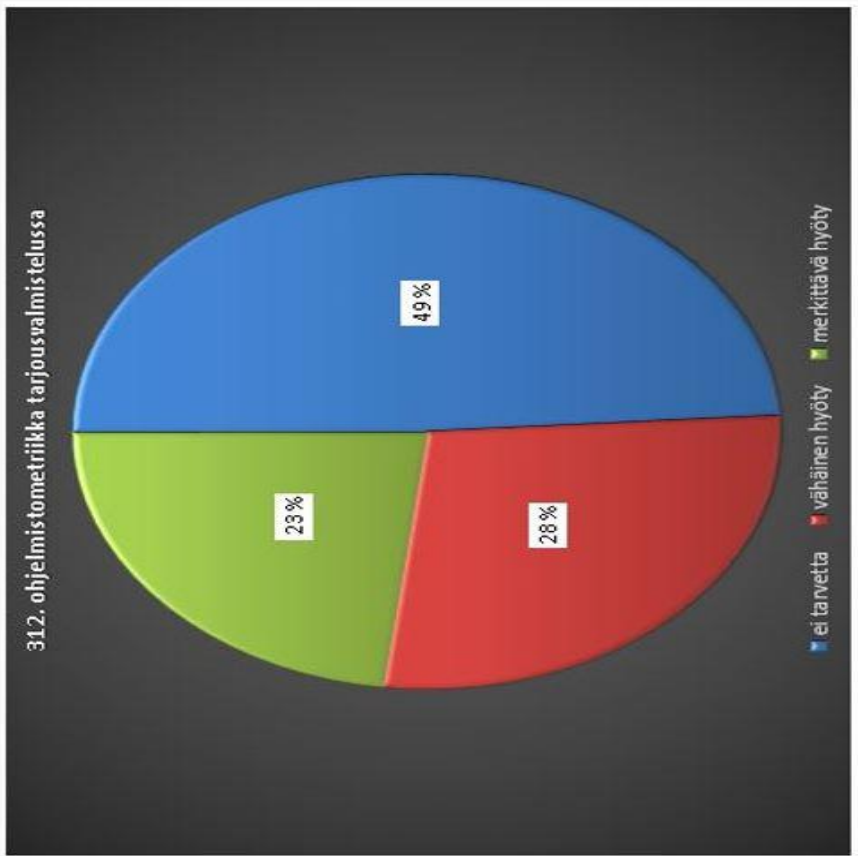
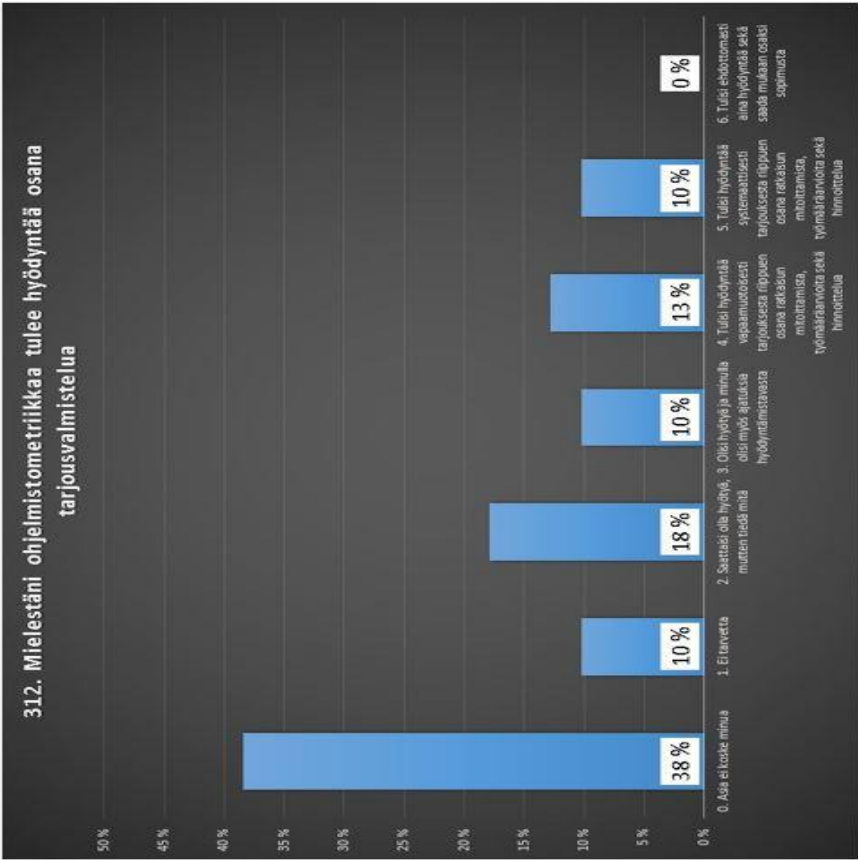
0. Asia ei koske minua	13 %	5	62 %	ei tarvetta tai tarpeeton (0-1)
1. Ei tarvitse	49 %	19		
2. Tarvitsen manuaalisen perustyövälineen tietojen tulkintaan/seurantaan/päätöksen tekoon	3 %	1	5 %	manuaalinen työväline (2-3)
3. Tarvitsen manuaalisen perustyövälineen tietojen tuottamiseksi	3 %	1		
4. Tarvitsen automaattisen työvälineen tietojen tulkintaan/seurantaan/päätöksen tekoon	3 %	1	33 %	automaattinen työväline (4-6)
5. Tarvitsen automaattisen työvälineen tietojen tuottamiseen	3 %	1		
6. Tarvitsen automaattisen ammattikäyttöön tarkoitetun työvälineen	28 %	11		



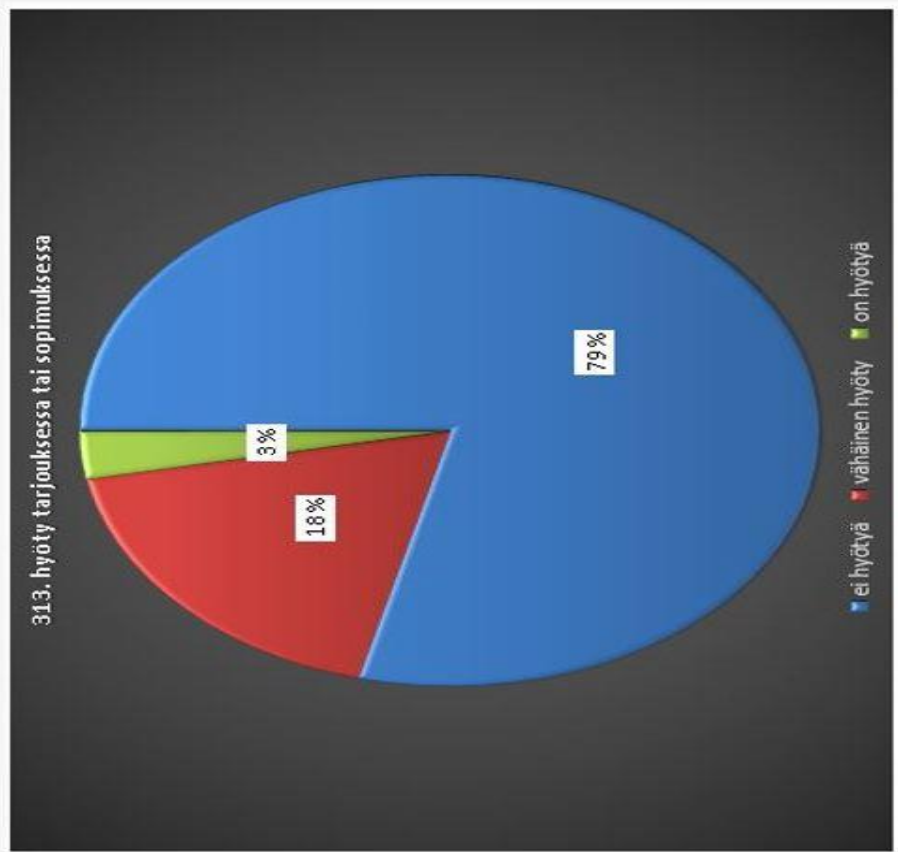
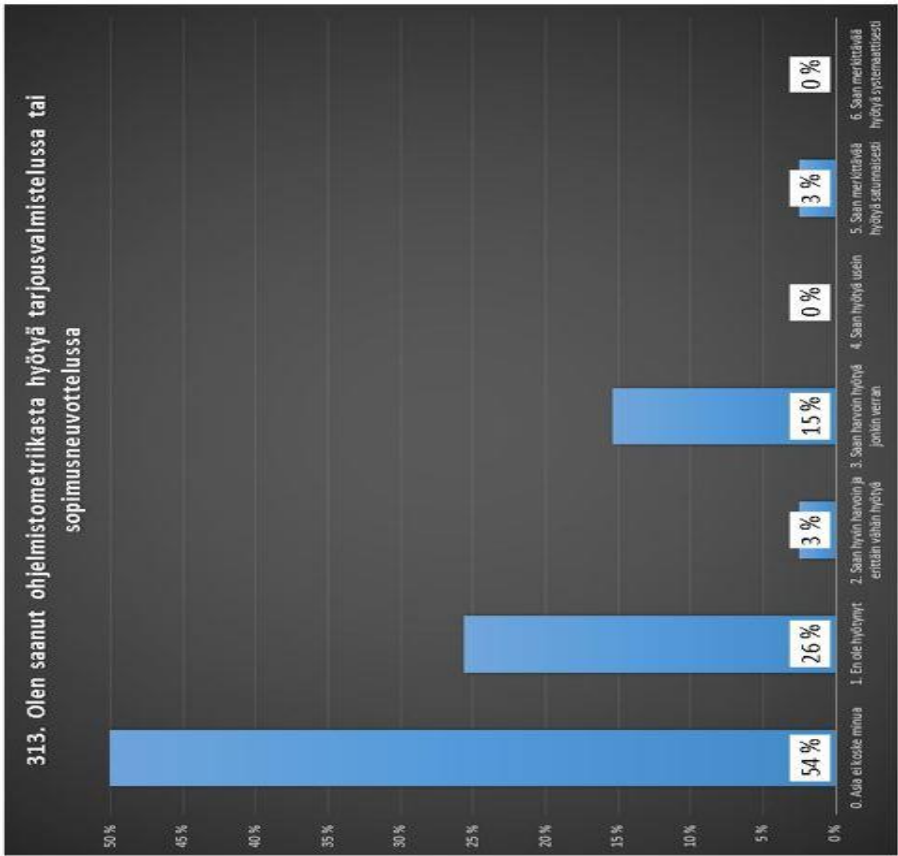
0. Asia ei koske minua	26 %	10	38 %	ei auta tai tarpeeton (0-1)
1. Ei auta	13 %	5		
2. Auttaisi hyvin vähän (ei merkitystä)	10 %	4	21 %	vähäinen apu (2-3)
3. Auttaisi jossain määrin	10 %	4		
4. Auttaisi paljon	13 %	5	41 %	merkittävä apu (4-6)
5. Parantaisi nykyistä tilannetta merkittävästi	13 %	5		
6. Kriittinen tekijä asetun laatuvaatteen sekä asiakaspauksen osalta	15 %	6		



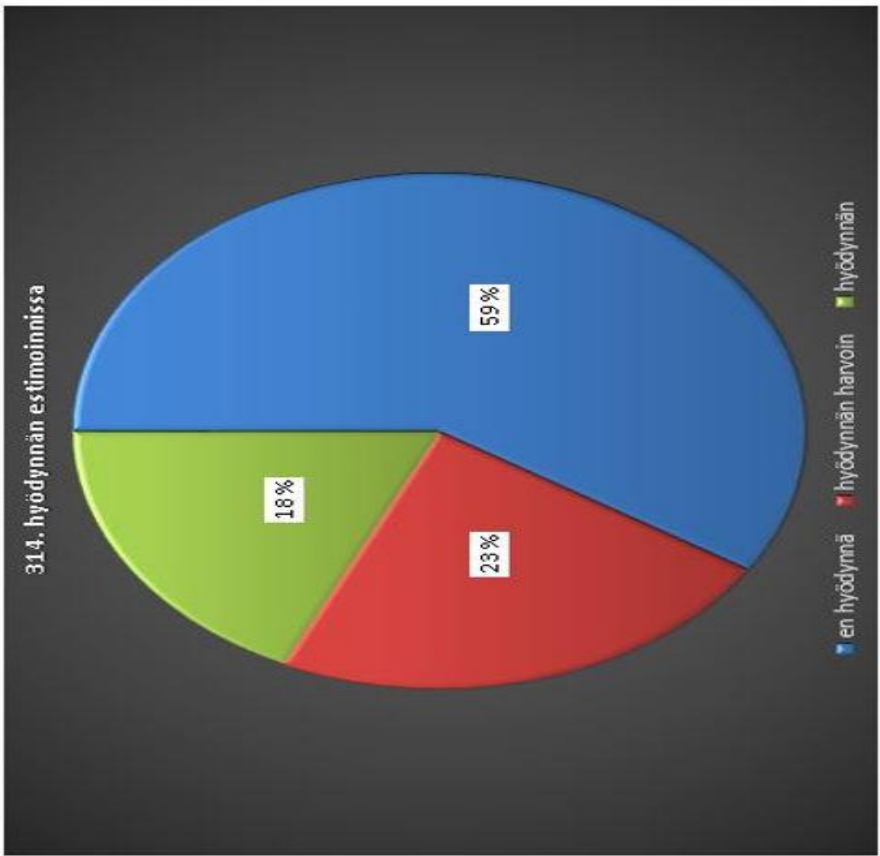
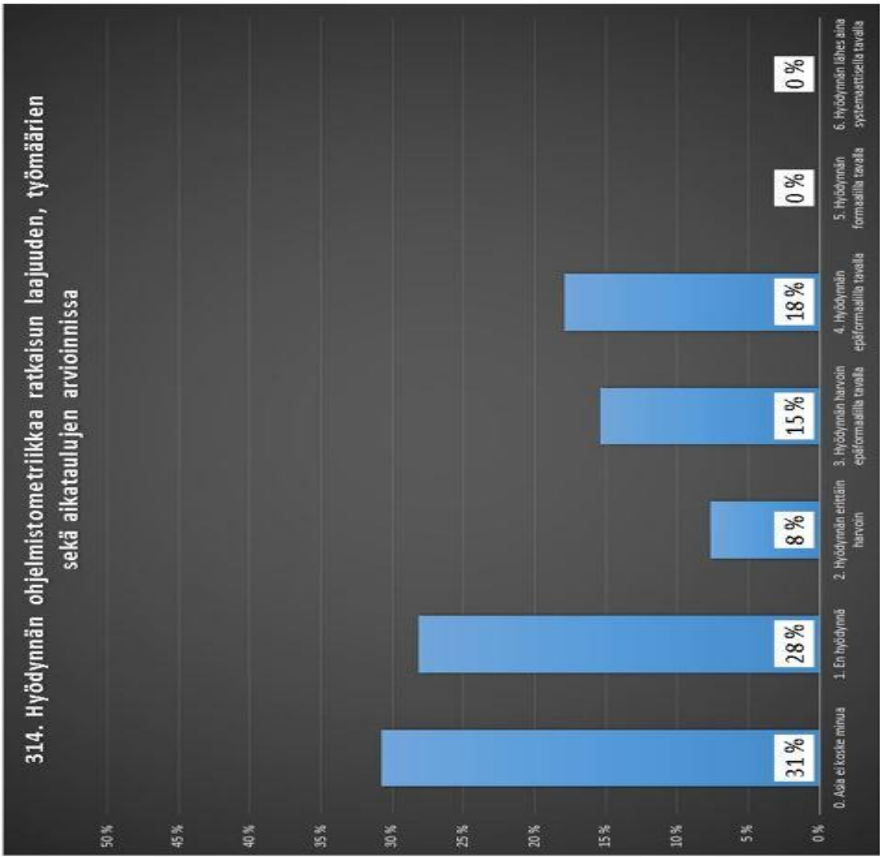
0. Asia ei koske minua	41 %	16	69 %	ei ole nähnyt (0-1)
1. En ole nähnyt tai kuullut	28 %	11		
2. Olen nähnyt toiveena	10 %	4	18 %	on nähnyt (2-3)
3. Olen nähnyt työohjeen tai toimintamallin muodossa	8 %	3		
4. Olen nähnyt osana laadun ohjausta	8 %	3	13 %	olennainen tekijä (4-6)
5. Olen nähnyt vaatimuksena	5 %	2		
6. Olen nähnyt ehdottomana vaatimuksena	0 %	0		



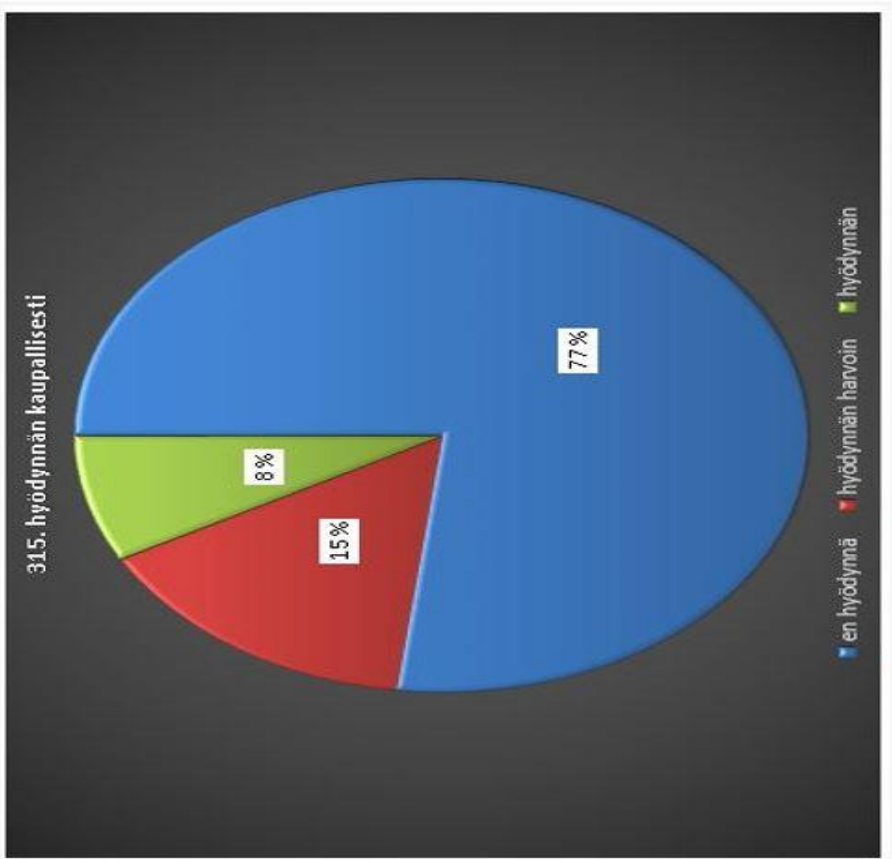
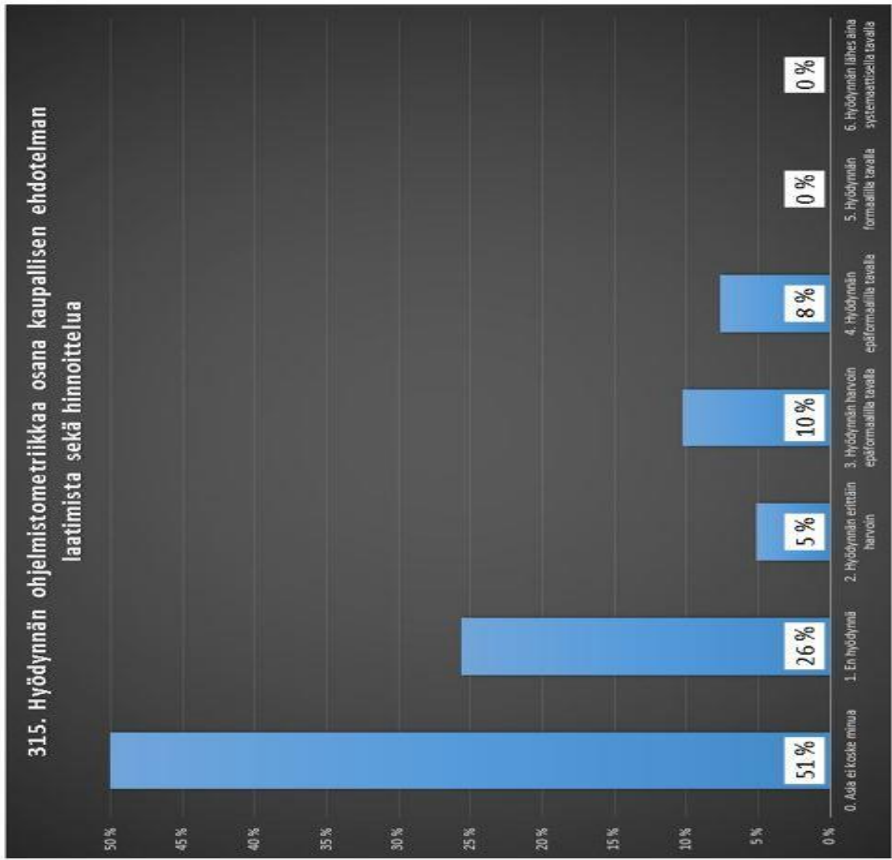
0. Asia ei koske minua	38 %	15	49 %	ei tarvetta (0-1)
1. Ei tarvetta	10 %	4		
2. Saattaisi olla hyödyä, mutta tiedä mitä hyödyntämistä	18 %	7	28 %	kannattaisi hyödyntää (2-3)
3. Olet hyötynyt ja minulla on myös ajatuksia hyödyntämisestä	10 %	4		
4. Tulisi hyödyntää vegaanuuksien tarjouksesta riippuen osana ratkaisun mittaamista, työmääräarvioita sekä hinnoittelua	13 %	5	23 %	tulee hyödyntää (4-6)
5. Tulisi hyödyntää systemaattisesti tarjouksesta riippuen osana ratkaisun mittaamista, työmääräarvioita sekä hinnoittelua	10 %	4		
6. Tulisi ehdottomasti aina hyödyntää sekä saada mukaan osaksi sopimusta	0 %	0		



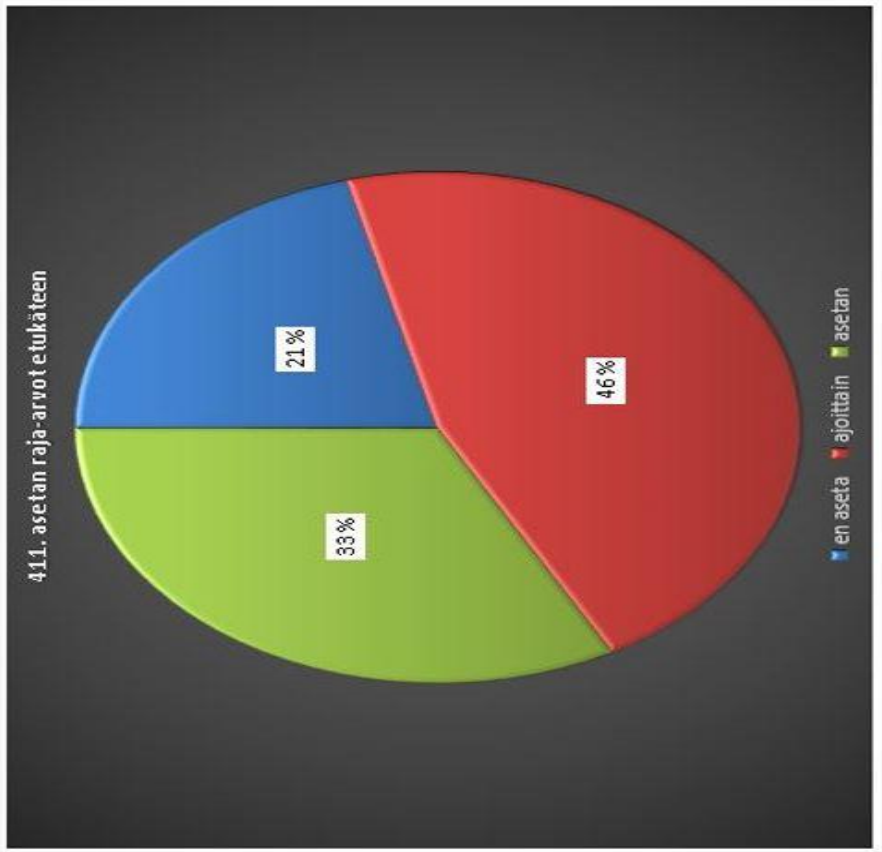
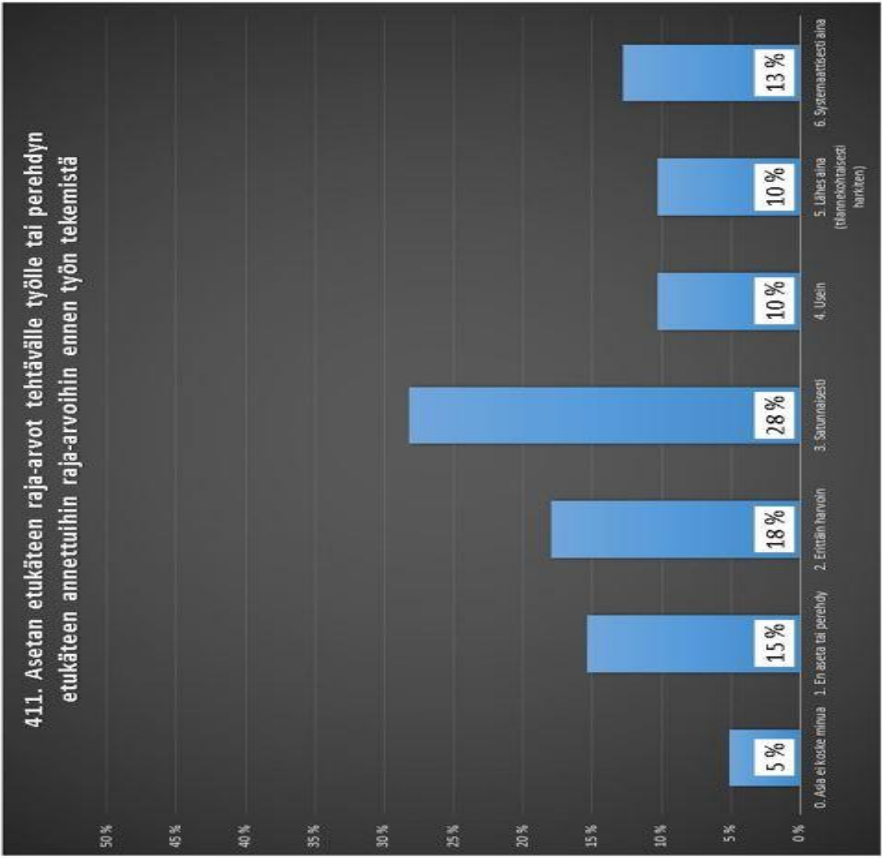
0. Asia ei koske minua	54 %	21	79 %	ei hyötyä (0-1)
1. En ole hyötynyt	26 %	10		
2. Saan hyvin harvoin ja erittäin vähän hyötyä	3 %	1	18 %	harvoin hyötyä (2-3)
3. Saan harvoin hyötyä jonkin verran	15 %	6		
4. Saan hyötyä usein	0 %	0	3 %	on hyötyä (4-6)
5. Saan merkittävästi hyötyä satunnaisesti	3 %	1		
6. Saan merkittävästi hyötyä systemaattisesti	0 %	0		



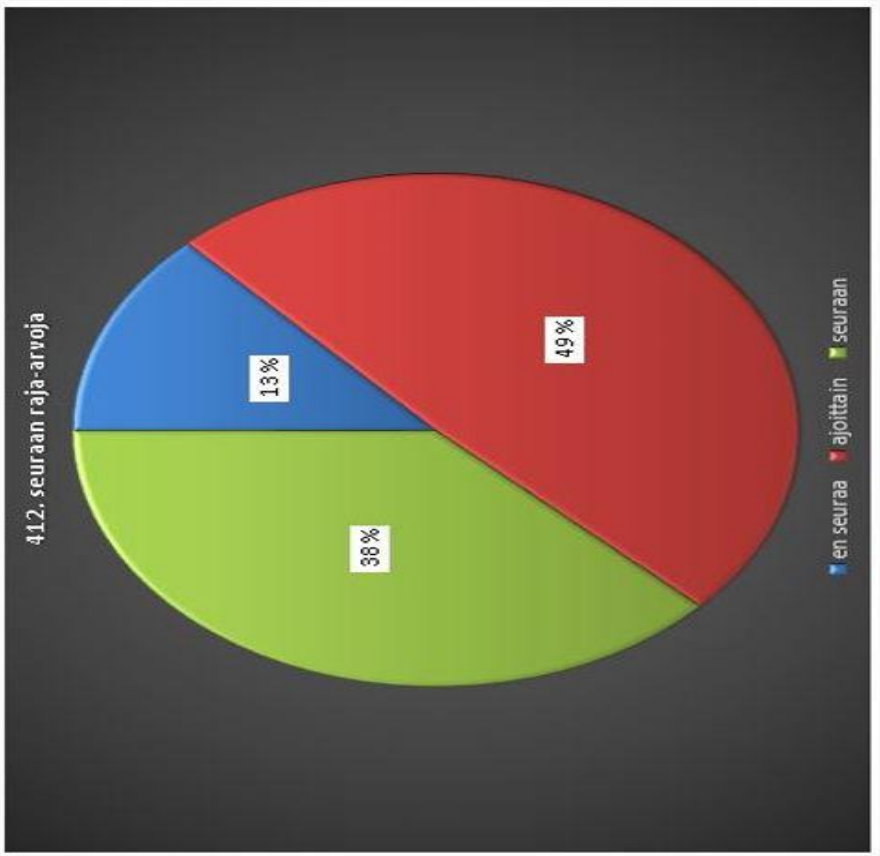
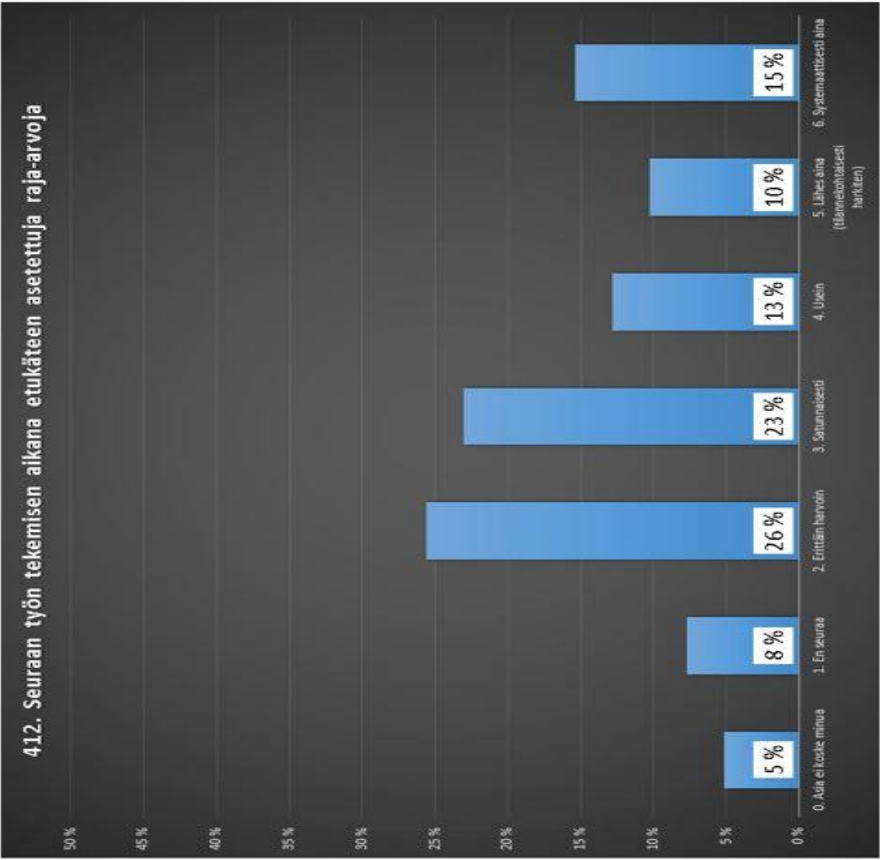
0. Asia ei koske minua	31 %	12	59 %	en hyödynnä (0-1)
1. En hyödynnä	28 %	11		
2. Hyödynnän erittäin harvoin	8 %	3	23 %	hyödynnän harvoin (2-3)
3. Hyödynnän harvoin epäformaalisella tavalla	15 %	6		
4. Hyödynnän epäformaalisella tavalla	18 %	7	18 %	hyödynnän (4-6)
5. Hyödynnän formaalisella tavalla	0 %	0		
6. Hyödynnän lähes aina systemaattisella tavalla	0 %	0		



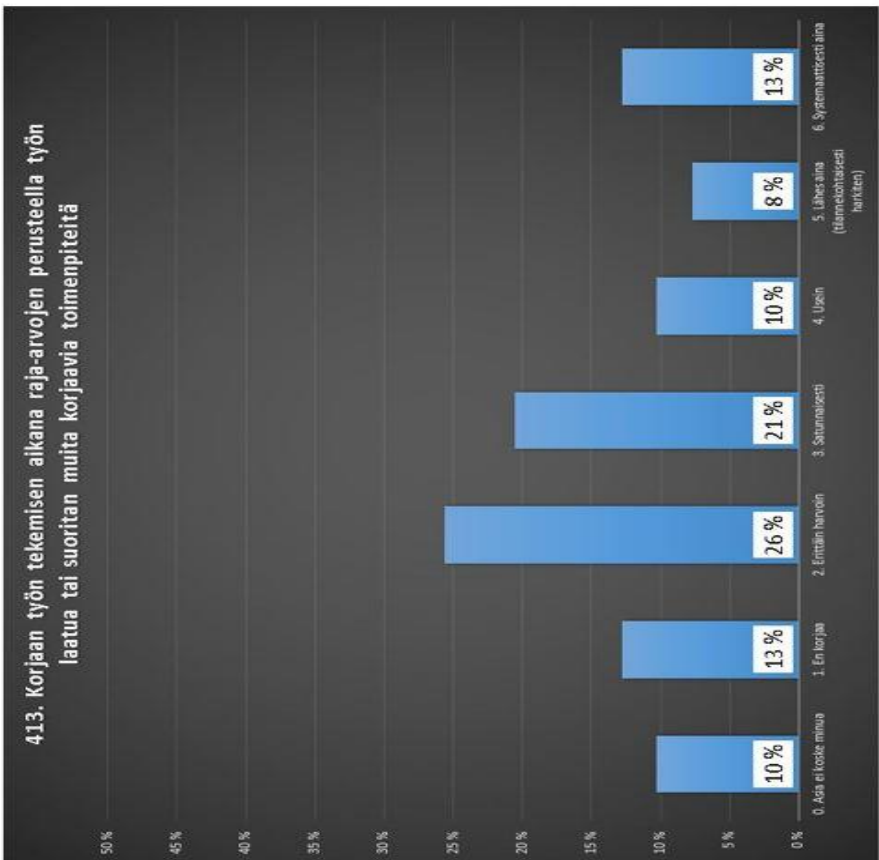
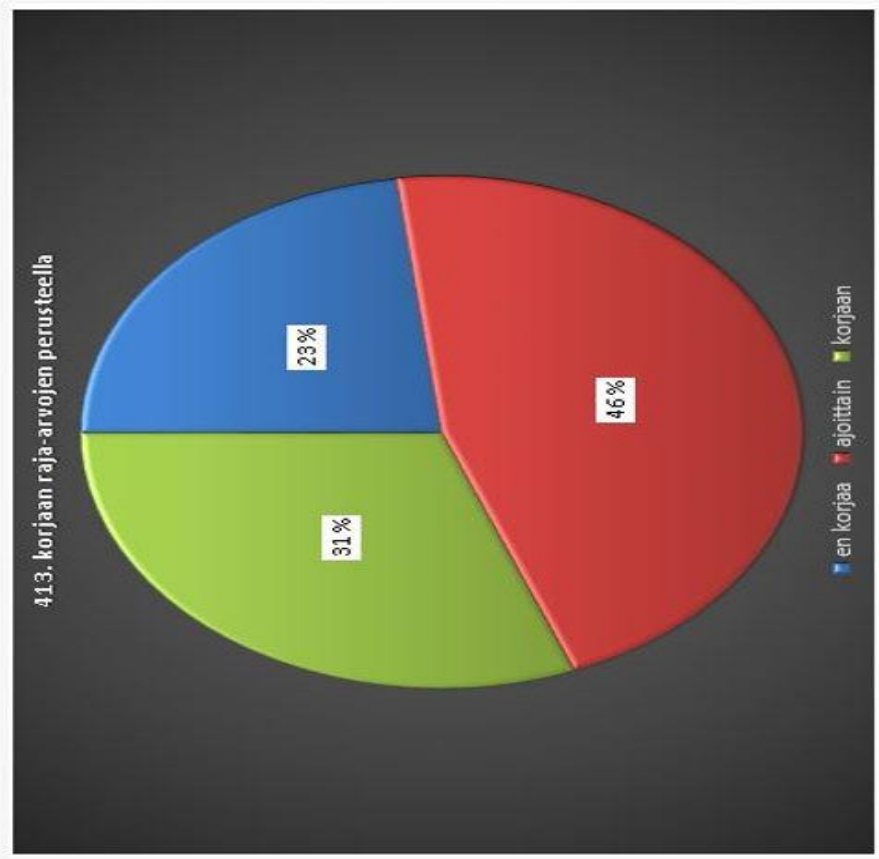
0. Asia ei koske minua	51 %	20	77 %	en hyödynnä (0-1)
1. En hyödynnä	26 %	10		
2. Hyödynnän erittäin harvoin	5 %	2	15 %	hyödynnän harvoin (2-3)
3. Hyödynnän harvoin epäformaalisella tavalla	10 %	4		
4. Hyödynnän epäformaalisella tavalla	8 %	3	8 %	hyödynnän (4-6)
5. Hyödynnän formaalisella tavalla	0 %	0		
6. Hyödynnän lähes aina systemaattisella tavalla	0 %	0		



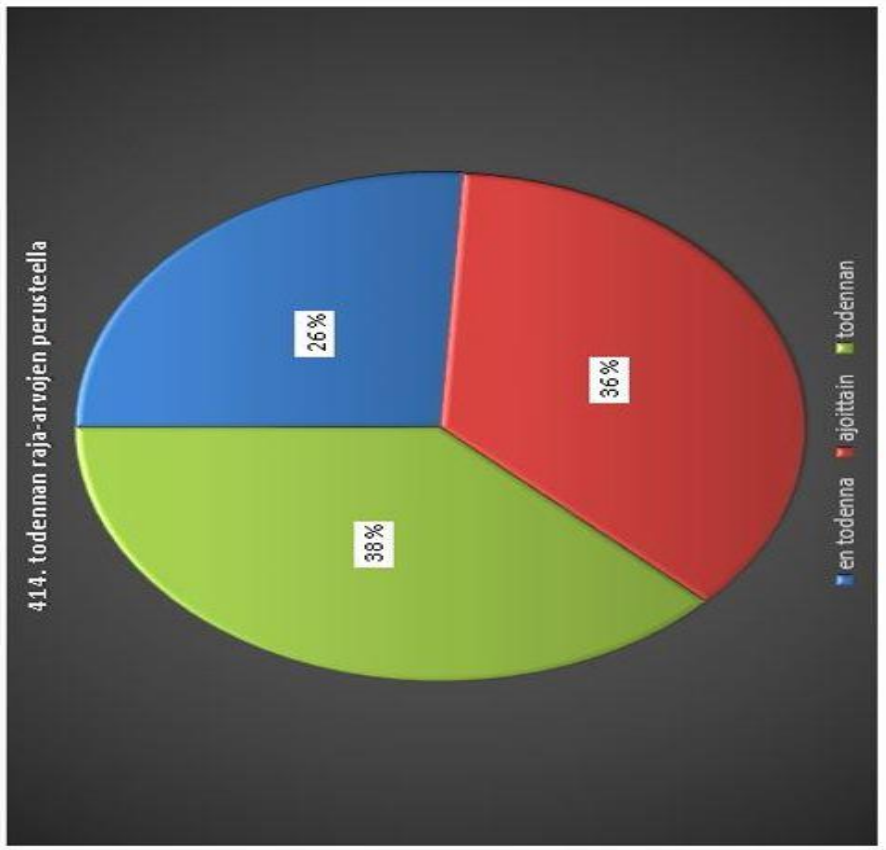
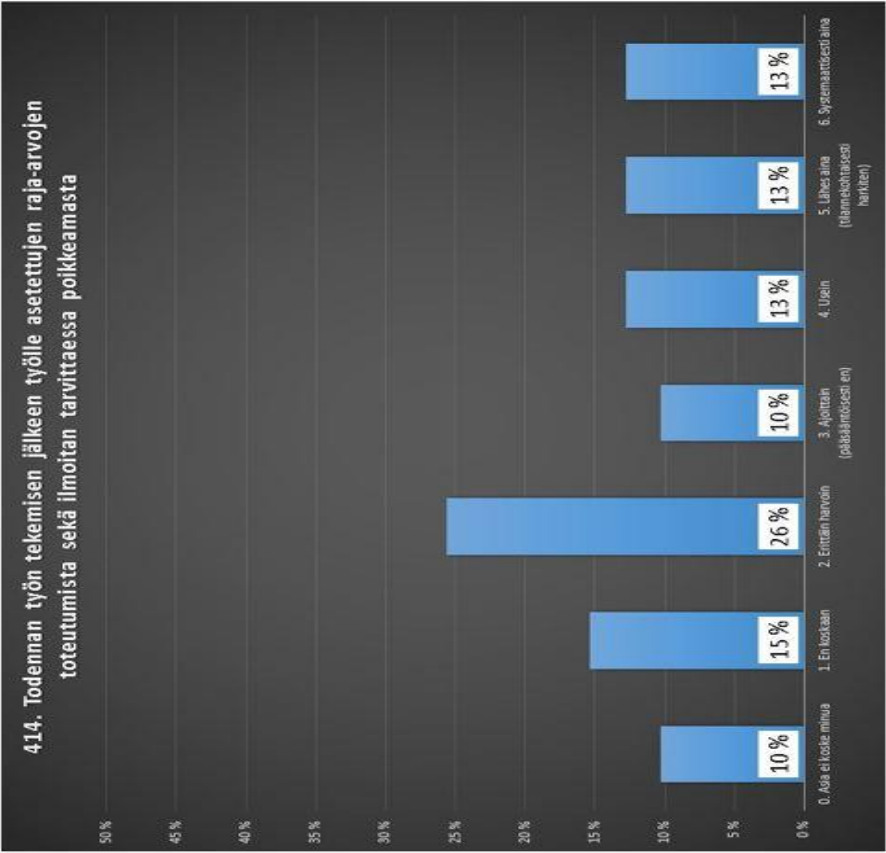
0. Asia ei koske minua	5 %	2	21 %	en aseta (0-1)
1. En aseta tai perehdy	15 %	6		
2. Erittäin harvoin	18 %	7	46 %	ajoittain (2-3)
3. Satunnaisesti	28 %	11		
4. Usein	10 %	4	33 %	asetan (4-6)
5. Lähes aina (tilannekohtaisesti harkiten)	10 %	4		
6. Systemaattisesti aina	13 %	5		



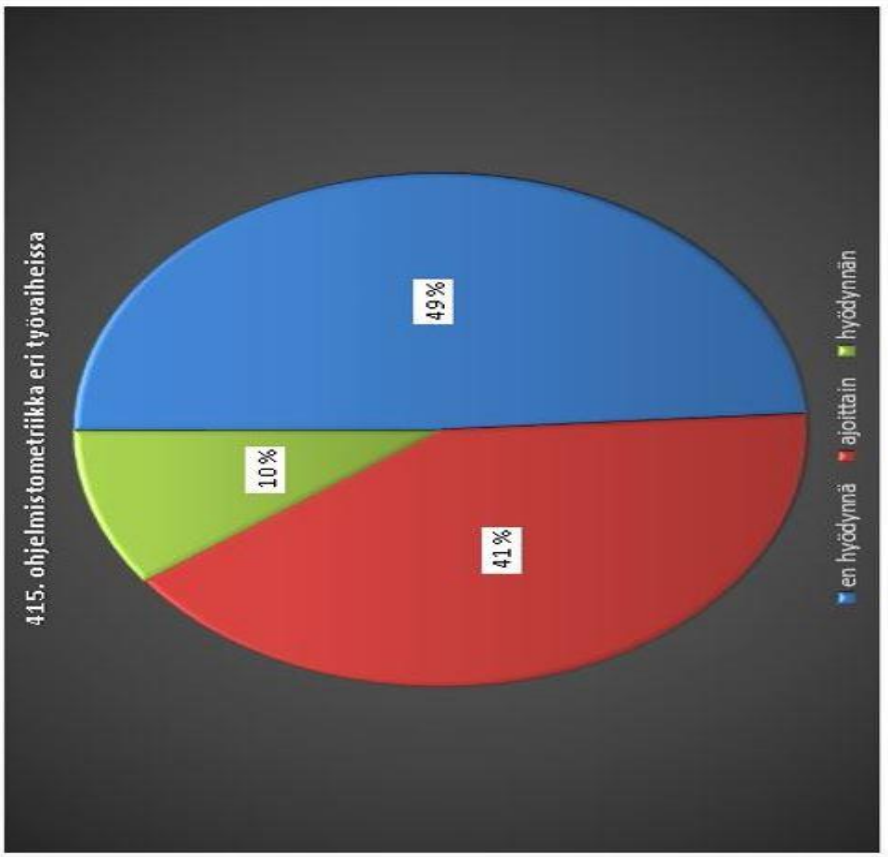
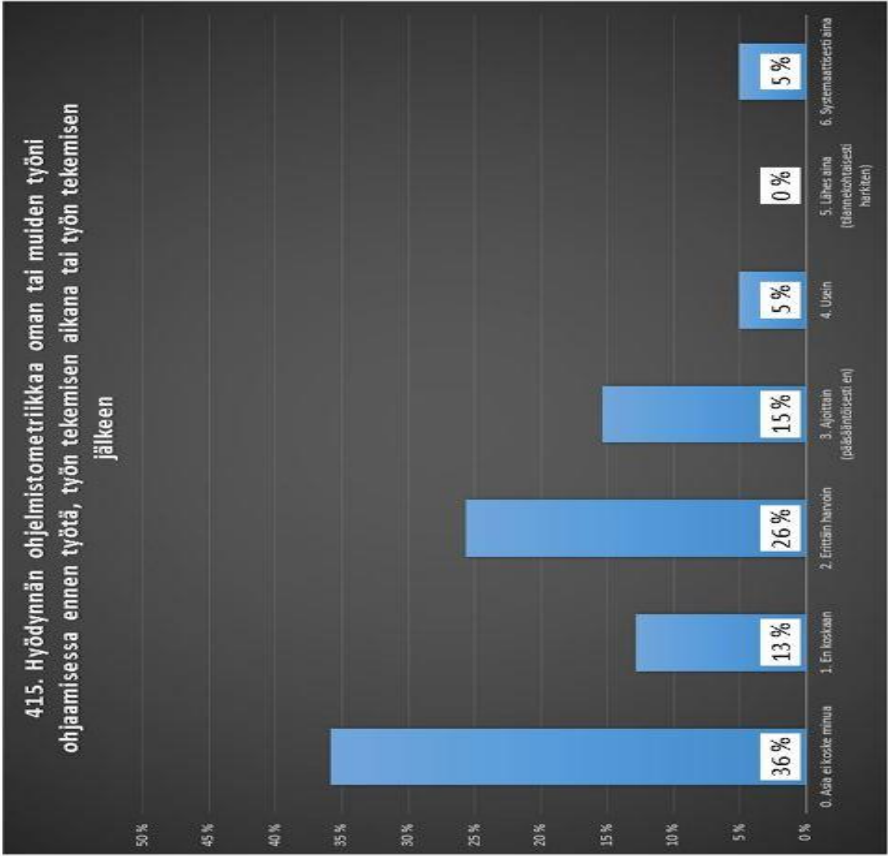
0. Asia ei koske minua	5 %	2	13 %	en seuraa (0-1)
1. En seuraa	8 %	3		
2. Erittäin harvoin	26 %	10	49 %	ajoitain (2-3)
3. Satunnaisesti	23 %	9		
4. Usein	13 %	5	38 %	seuraan (4-6)
5. Lähes aina (tilannekohtaisesti harkiten)	10 %	4		
6. Systemaattisesti aina	15 %	6		



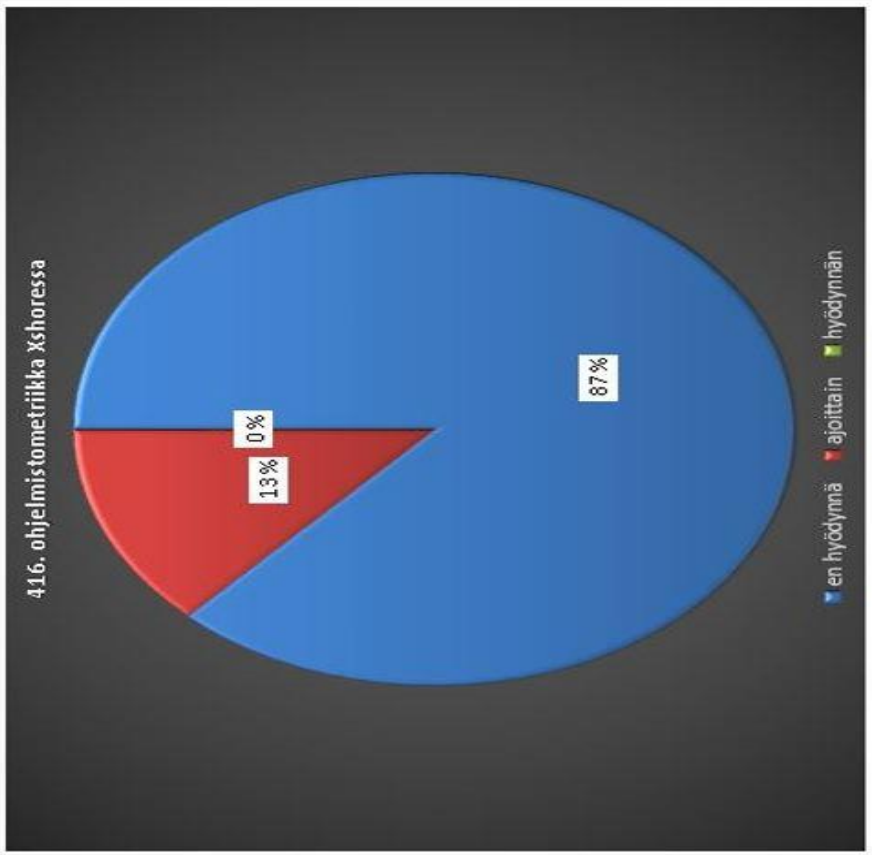
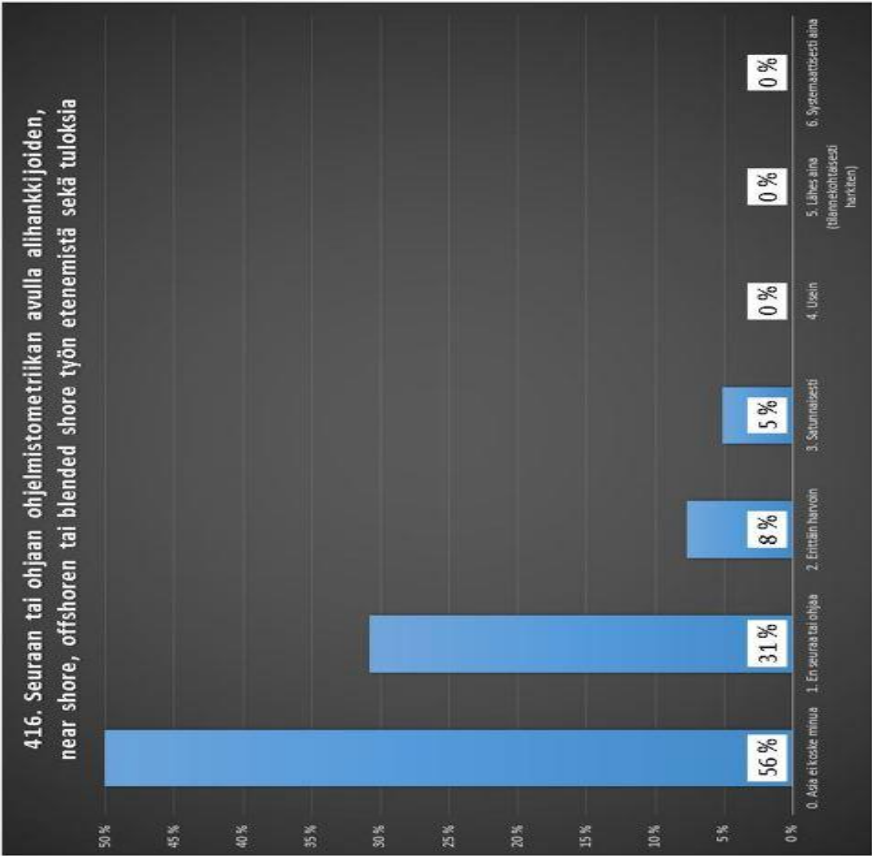
0. Asia ei koske minua	10 %	4	23 %	en korjaa (0-1)
1. En korjaa	13 %	5		
2. Erittäin harvoin	26 %	10	46 %	ajottain (2-3)
3. Satunnaisesti	21 %	8		
4. Usein	10 %	4	31 %	korjaan (4-6)
5. Lähes aina (tilannekohtaisesti harkiten)	8 %	3		
6. Systemaattisesti aina	13 %	5		



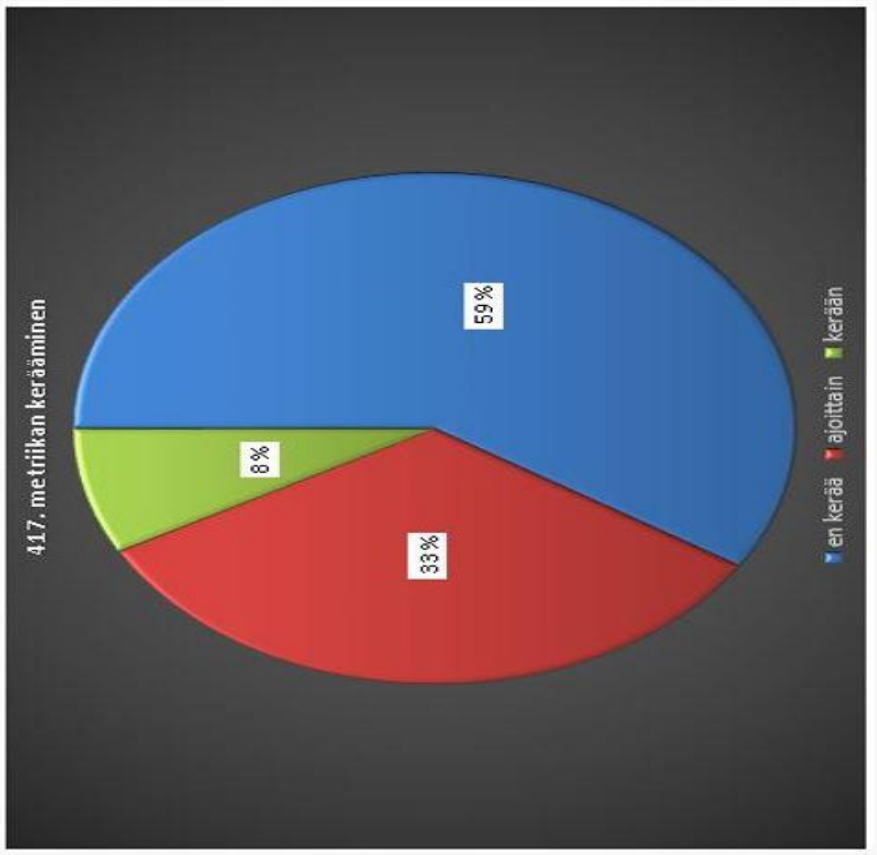
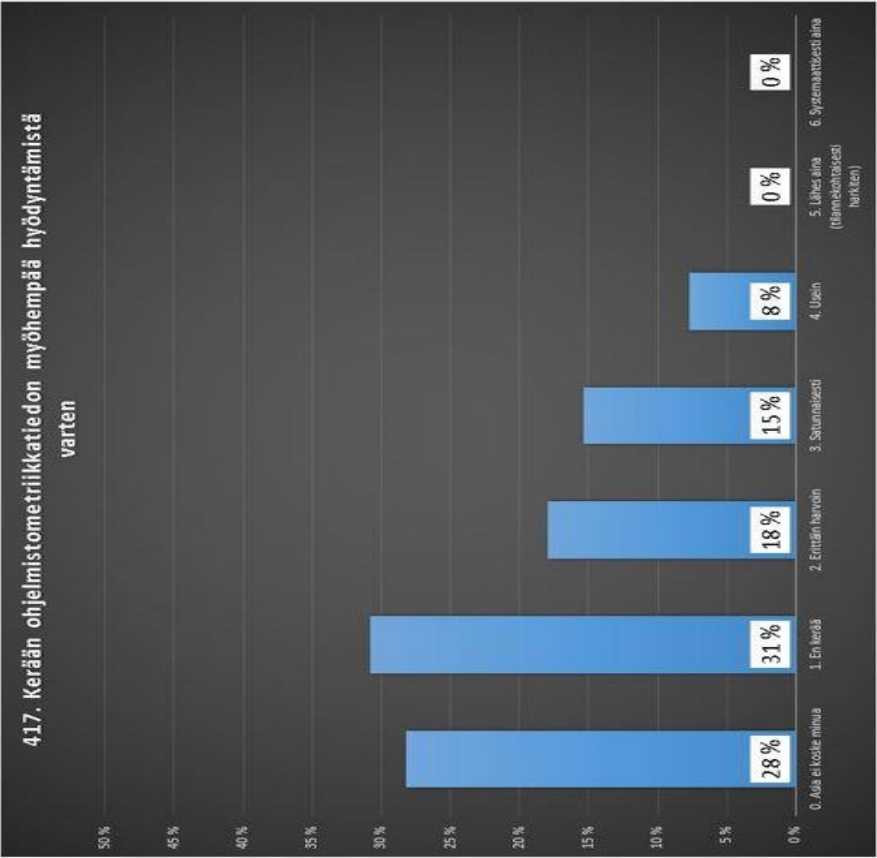
0. Asia ei koske minua	10 %	4	26 %	en todenna (0-1)
1. En koskaan	15 %	6		
2. Erittäin harvoin	26 %	10	36 %	ajottain (2-3)
3. Ajoittain (pääsääntöisesti en)	10 %	4		
4. Usein	13 %	5	38 %	todennan (4-6)
5. Lähes aina (tilannekohtaisesti harkiten)	13 %	5		
6. Systemaattisesti aina	13 %	5		



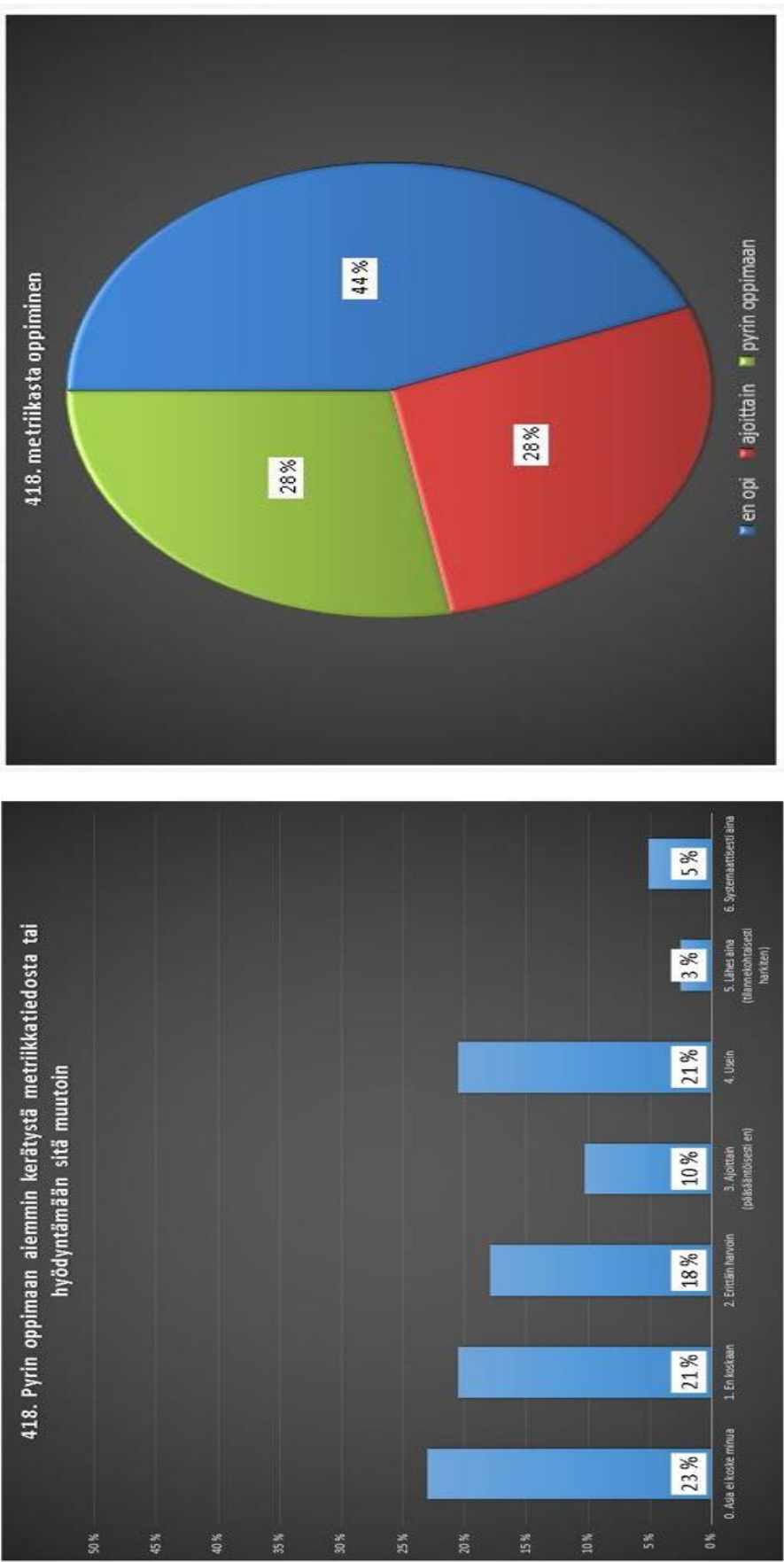
0. Asia ei koske minua	36 %	14	49 %	en hyödynnä (0-1)
1. En koskaan	13 %	5		
2. Erittäin harvoin	26 %	10	41 %	ajottain (2-3)
3. Ajoittain (pääsääntöisesti en)	15 %	6		
4. Usein	5 %	2	10 %	hyödynnän (4-6)
5. Lähes aina (tilannekohtaisesti harkiten)	0 %	0		
6. Systemaattisesti aina	5 %	2		



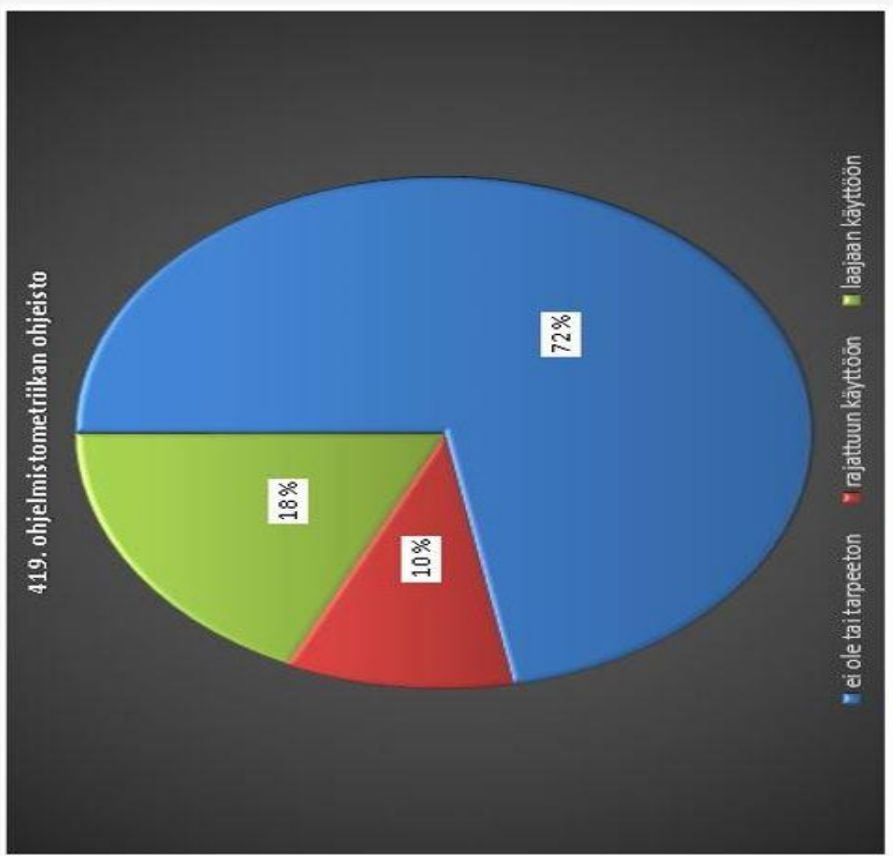
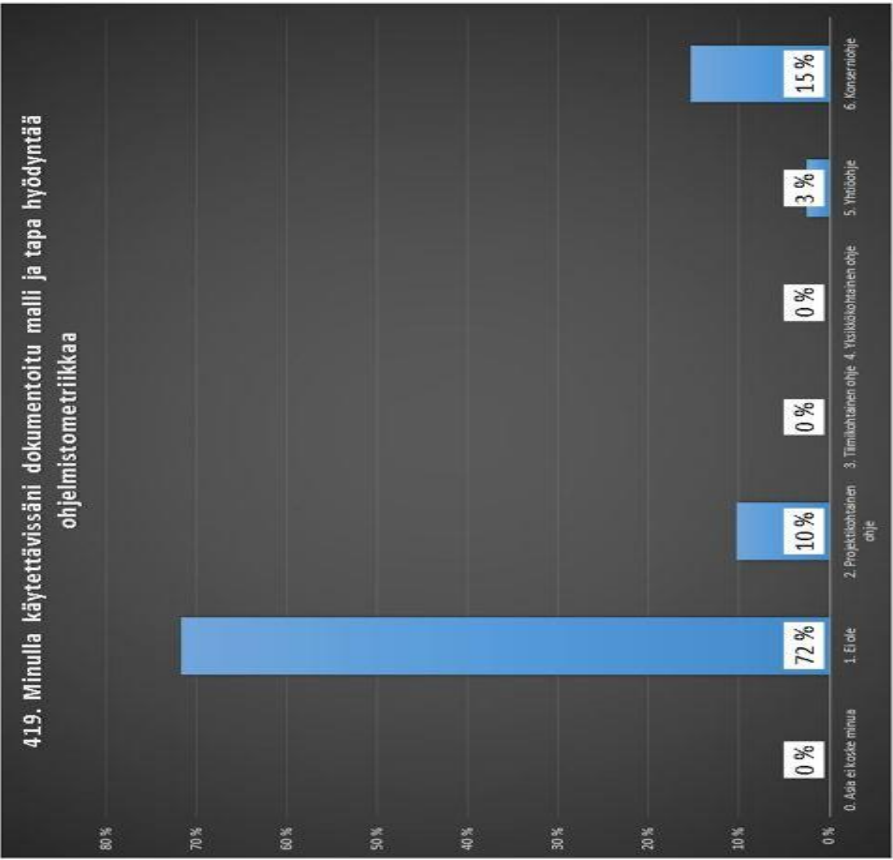
0. Asia ei koske minua	56 %	22	87 %	en hyödynnä (0-1)
1. En seuraa tai ohjaa	31 %	12		
2. Erittäin harvoin	8 %	3	13 %	ajoittain (2-3)
3. Satunnaisesti	5 %	2		
4. Usein	0 %	0	0 %	hyödynnän (4-6)
5. Lähes aina (tilannekohtaisesti harkiten)	0 %	0		
6. Systemaattisesti aina	0 %	0		



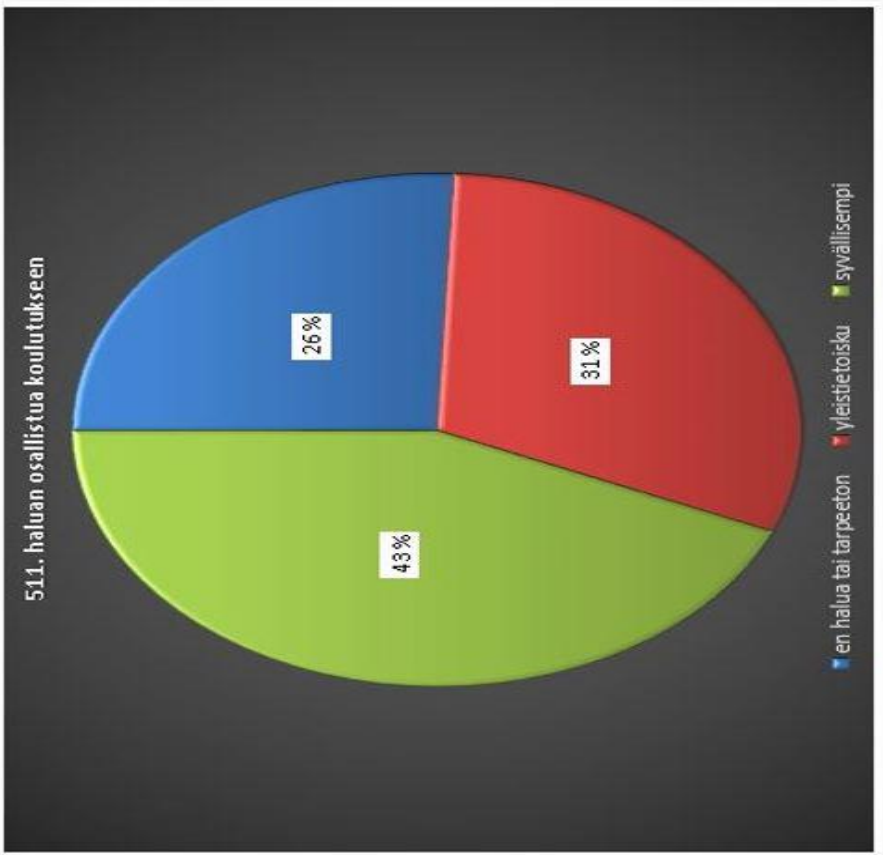
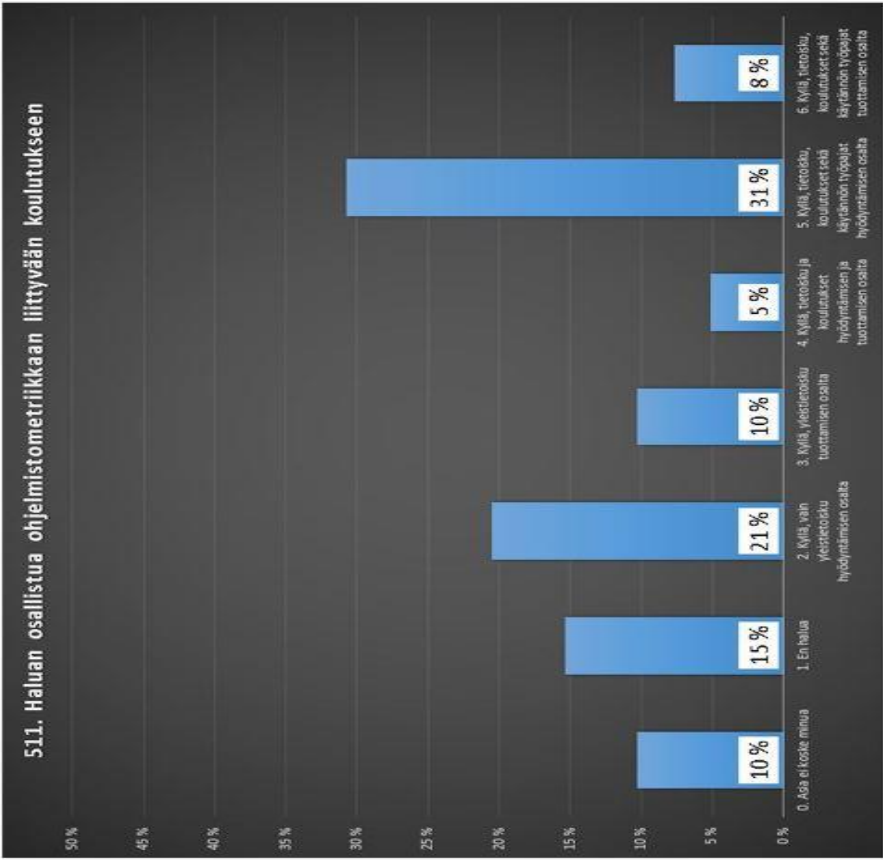
0. Asia ei koske minua	28 %	11	59 %	en kerää (0-1)
1. En kerää	31 %	12		
2. Erittäin harvoin	18 %	7	33 %	ajoitain (2-3)
3. Satunnaisesti	15 %	6		
4. Usein	8 %	3	8 %	kerään (4-6)
5. Lähes aina (tilannekohtaisesti harkiten)	0 %	0		
6. Systemaattisesti aina	0 %	0		



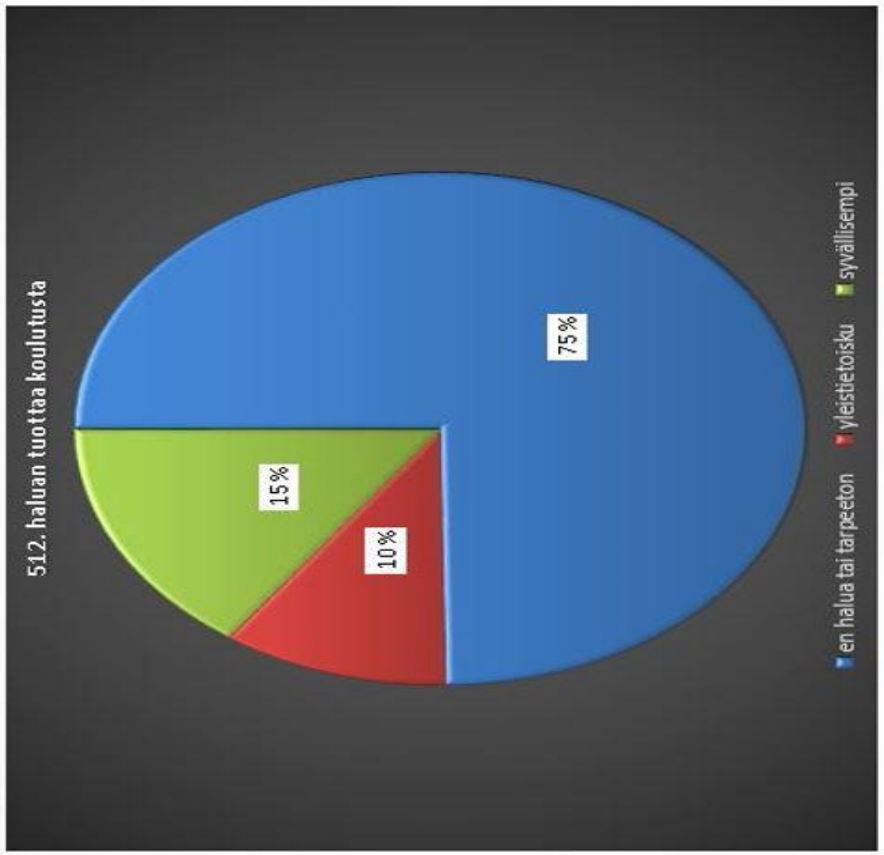
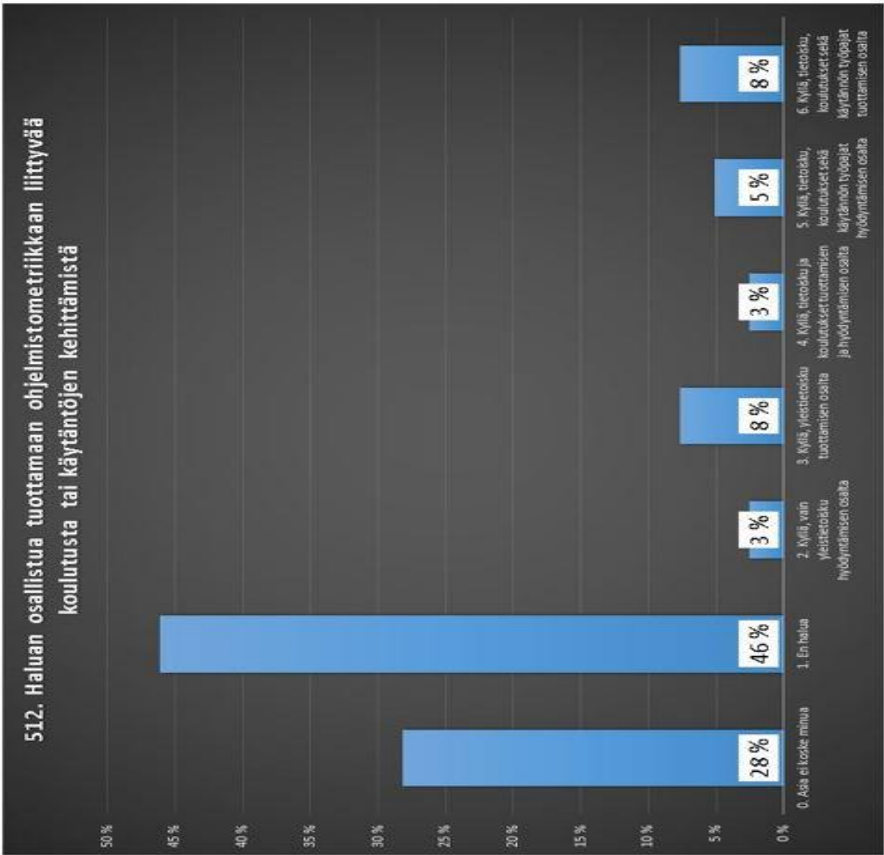
0. Asia ei koske minua	23 %	9	44 %	en opi (0-1)
1. En koskaan	21 %	8		
2. Erittäin harvoin	18 %	7	28 %	ajottain (2-3)
3. Ajoittain (pääsääntöisesti en)	10 %	4		
4. Usein	21 %	8	28 %	pyrin oppimaan (4-6)
5. Lähes aina (tilannekohtaisesti harkiten)	3 %	1		
6. Systemaattisesti aina	5 %	2		



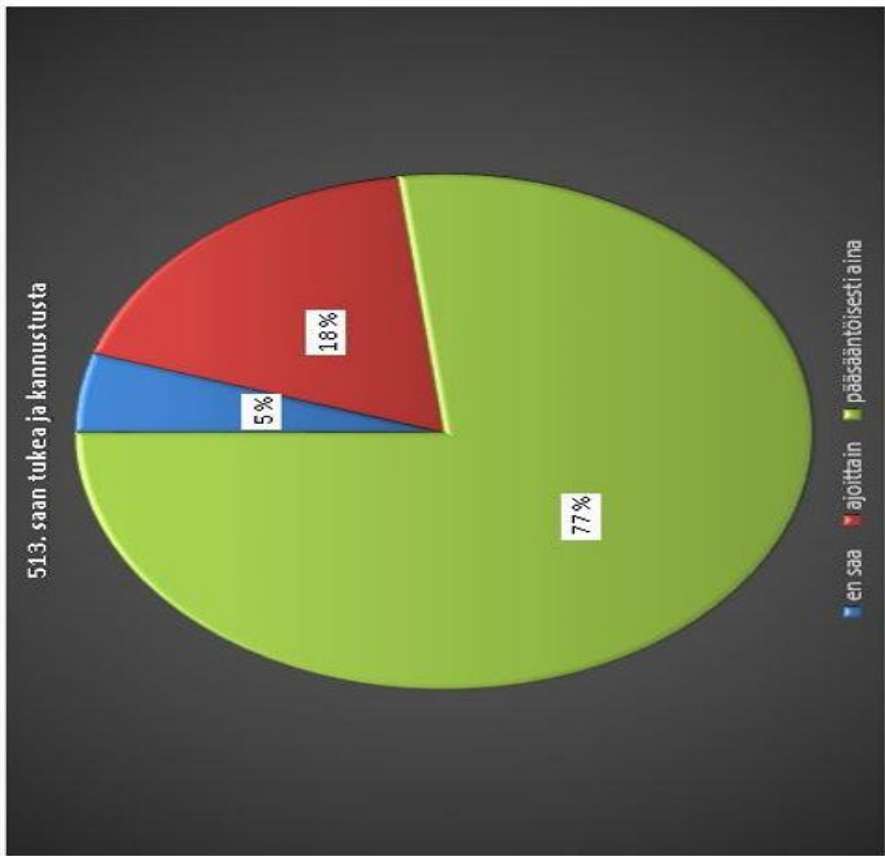
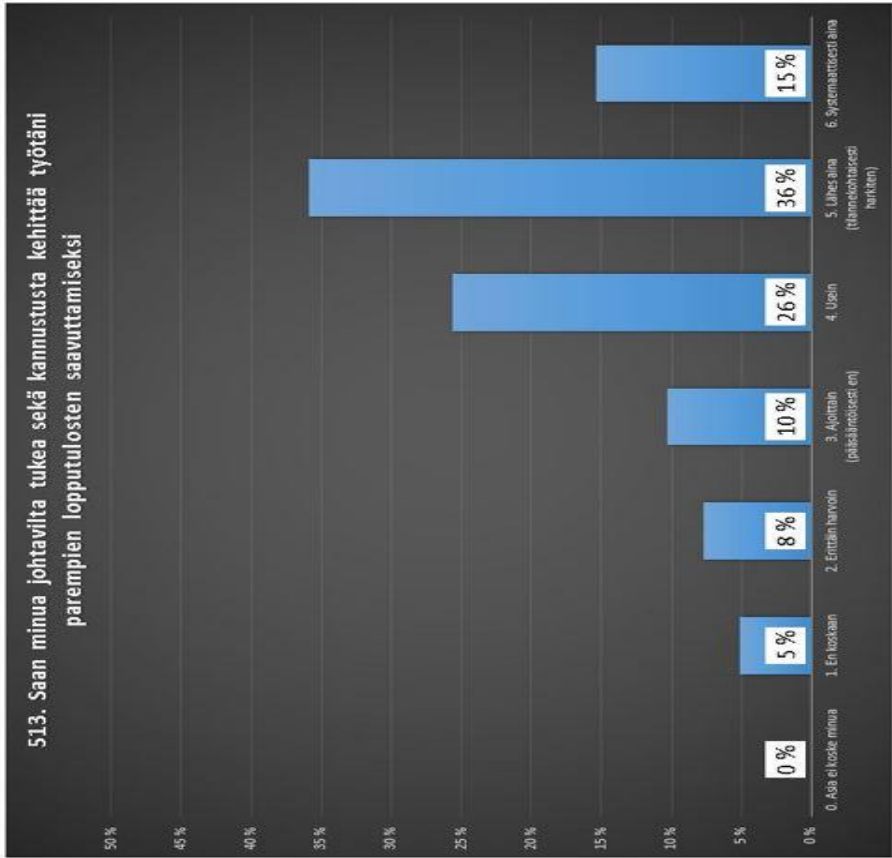
0. Asia ei koske minua	0 %	0	72 %	ei ole tai tarpeeton (0-1)
1. Ei ole	72 %	28		
2. Projektikohtainen ohje	10 %	4	10 %	rajattuun käyttöön (2-3)
3. Tiimikohtainen ohje	0 %	0		
4. Yksikkökohtainen ohje	0 %	0	18 %	laajaan käyttöön (4-6)
5. Yhtiöohje	3 %	1		
6. Konserniohje	15 %	6		



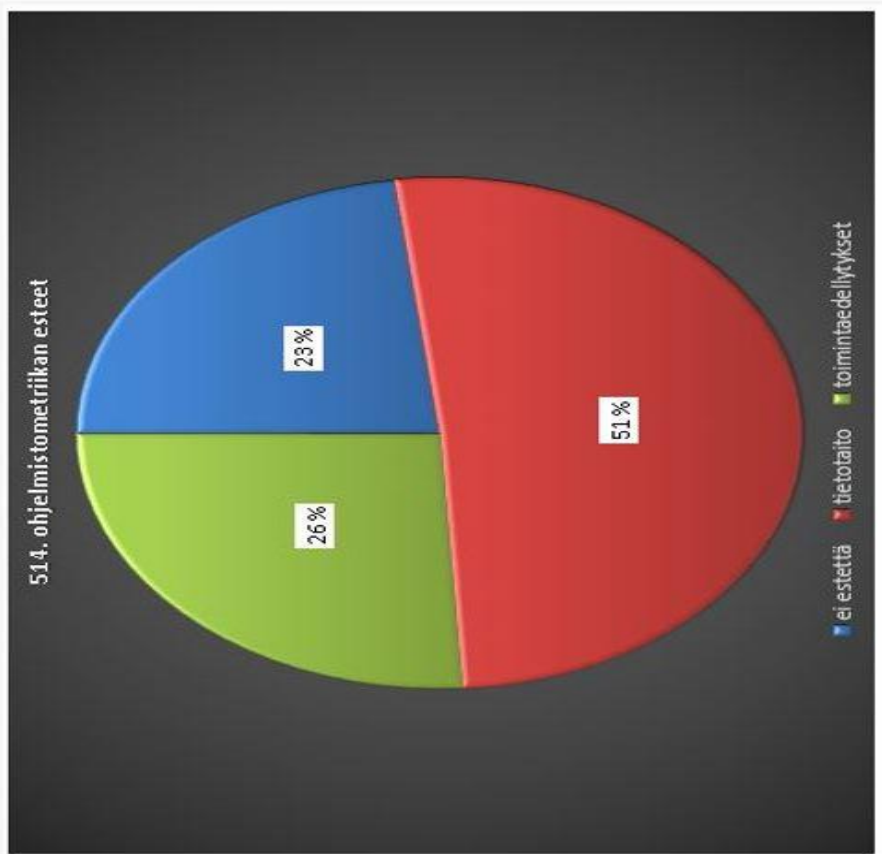
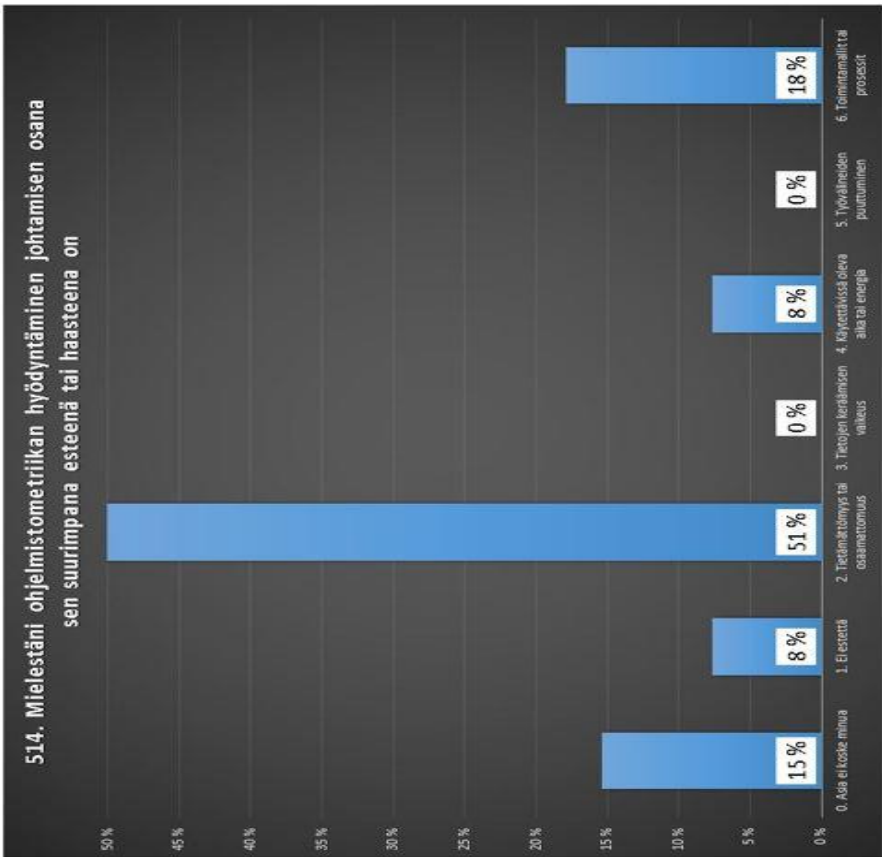
0. Asia ei koske minua	10 %	4	26 %	en halua tai tarpeeton (0-1)
1. En halua	15 %	6		
2. Kyllä, vain yleistietoisku hyödyntämisen osalta	21 %	8	31 %	yleistietoisku (2-3)
3. Kyllä, yleistietoisku tuottamisen osalta	10 %	4		
4. Kyllä, tietoisuuden ja koulutuksen hyödyntämisen osalta	5 %	2	44 %	syvällisempi (4-6)
5. Kyllä, tietoisuuden, koulutuksen sekä käytännön työpajat hyödyntämisen osalta	31 %	12		
6. Kyllä, tietoisuuden, koulutuksen sekä käytännön työpajat tuottamisen osalta	8 %	3		



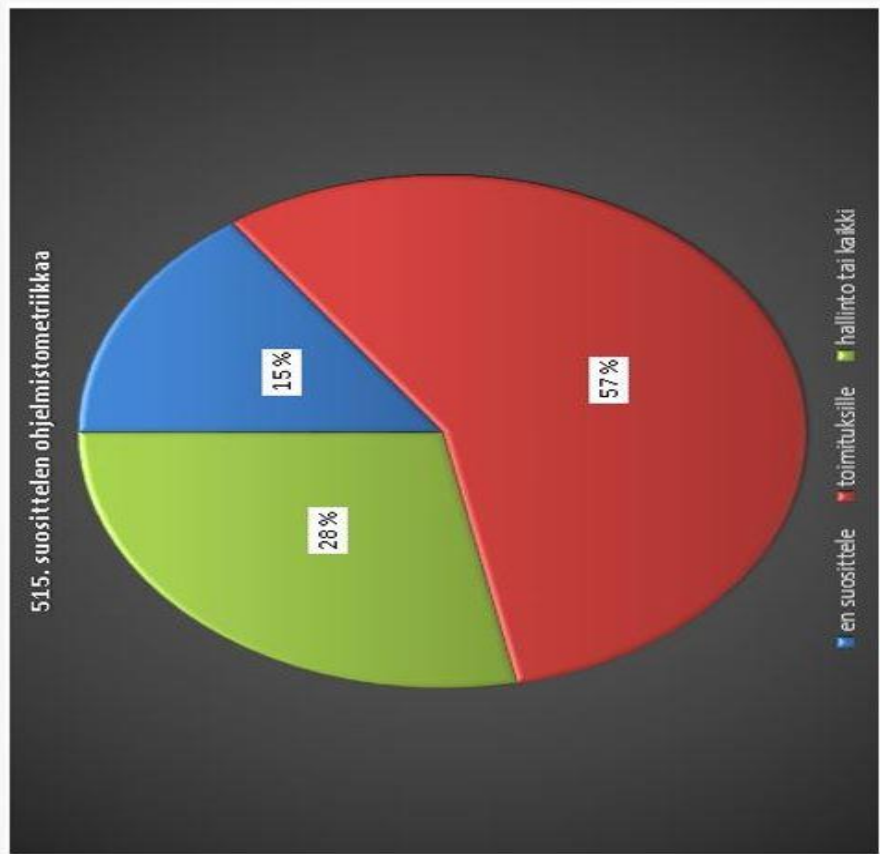
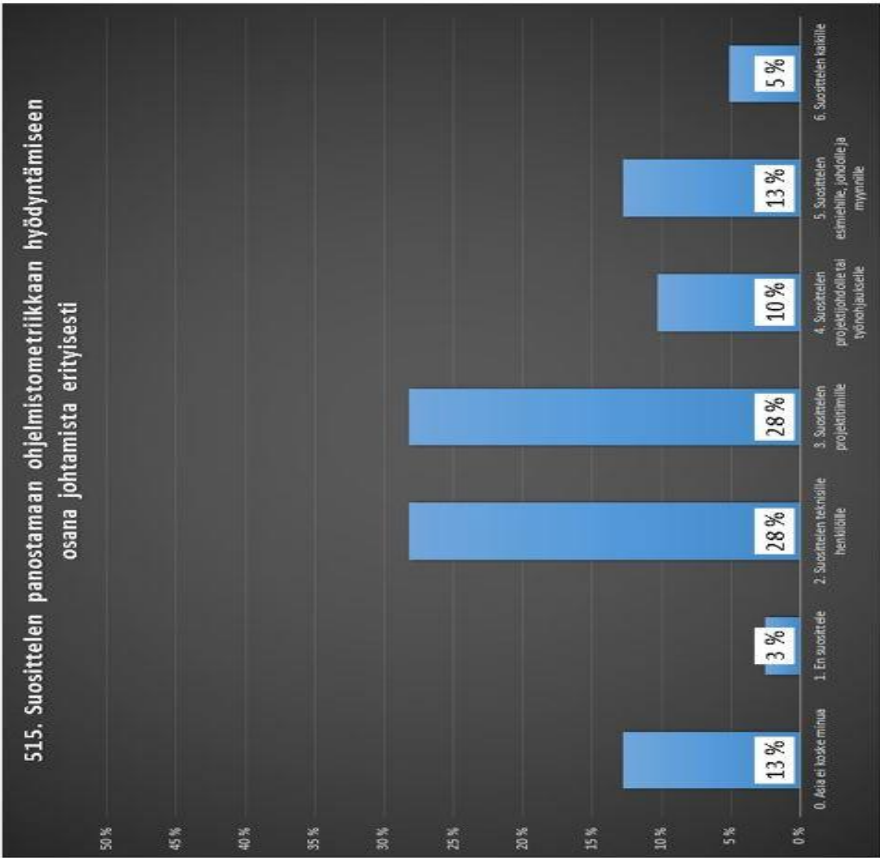
0. Asia ei koske minua	28 %	11	74 %	en halua tai tarpeeton (0-1)
1. En halua	46 %	18		
2. Kyllä, vain yleistietoisku hyödyntämisen osalta	3 %	1	10 %	yleistietoisku (2-3)
3. Kyllä, yleistietoisku tuottamisen osalta	8 %	3		
4. Kyllä, tietoisu- ja koulutukset tuottamisen ja hyödyntämisen osalta	3 %	1	15 %	syvällisempi (4-6)
5. Kyllä, tietoisu- ja koulutukset sekä käytännön työpajat hyödyntämisen osalta	5 %	2		
6. Kyllä, tietoisu- ja koulutukset sekä käytännön työpajat tuottamisen osalta	8 %	3		



0. Asia ei koske minua	0%	0	5%	en saa kannustusta tai tarpeeton (0-1)
1. En koskaan	5%	2		
2. Erittäin harvoin	8%	3	18%	ajottain (2-3)
3. Ajoittain (pääsääntöisesti en)	10%	4		
4. Usein	26%	10	77%	pääsääntöisesti aina (4-6)
5. Lähes aina (tilannekohtaisesti harkiten)	36%	14		
6. Systemaattisesti aina	15%	6		



0. Asia ei koske minua	15%	6	23%	ei estettä (0-1)
1. Ei estettä	8%	3		
2. Tietämättömyys tai osaamattomuus	51%	20	51%	tietotaito (2-3)
3. Tietojen keräämisen vaikeus	0%	0		
4. Käytettävissä oleva aika tai energia	8%	3	26%	toimintaedellytykset (4-6)
5. Työvälineiden puuttuminen	0%	0		
6. Toimintamallit tai prosessit	18%	7		



0. Asia ei koske minua	13 %	5	15 %	en suosittele (0-1)
1. En suosittele	3 %	1		
2. Suosittelemme teknisiille henkilöille	28 %	11	56 %	toimitukselle (2-3)
3. Suosittelemme projektitiimille	28 %	11		
4. Suosittelemme projektijohtajalle tai työohjaajalle	10 %	4	28 %	hallinto tai kaikki (4-6)
5. Suosittelemme esimiehille, johdolle ja myynnille	13 %	5		
6. Suosittelemme kaikille	5 %	2		